

C프로그래밍 및 실습

자동 의류 추천

최종 보고서

제출일자: 2023/12/24

제출자명: 나예준

제출자학번: 232867

1. 프로젝트 목표

1) 배경 및 필요성

근래 날씨가 추워지는 동향을 보이다가 갑자기 더운 날이 오는 바람에 옷을 평소처럼 입다가 너무 더워서 힘들어 하는 경험을 겪었고, 나를 비롯해 많은 주변인들도 힘들어 하는 모습을 보았다. 날씨의 변화와 의류는 큰 밀접성을 가지고 있지만 꼼꼼하지 못한 성격을 가진 사람들은 날씨를 크게 반영하지 않고 옷을 입고 나와서 봉변을 당하거나 날씨를 알아도 옷을 어떻게 입어야 할 지 모르는 사람들을 위해 이 프로그램을 고안하게 되었다.

2) 프로젝트 목표

(1) 오늘의 날씨와 (2) 사용자가 가지고 있고, 취향에 맞는 옷들을 입력 받으면 날씨와 사용자의 취향을 고려해서 옷을 추천해주는 것을 목표로 한다.

3) 차별점

날씨를 알려주는 기존 프로그램들이 많이 있고 의류를 추천해주는 프로그램들이 있지만 그것들을 통합해서 한번에 날씨 정보와 더불어 내게 맞는 옷을 추천 받을 수 있다. 또한 사용자가 가지고 있는 옷들을 기반으로 추천해주는 옷이기 때문에 구매를 유도하는 광고성 프로그램과 차별점이 있다.

2. 기능 계획

1) 기능 1 사용자에게 필요한 정보 입력 받기

- 사용자에게 오늘의 날씨와 사용자가 가지고 있는 옷을 입력 받는다.

프로그램의 제작 의도는 날씨에 맞는 옷을 자동으로 추천해줘서 편리함을 제공하는 것인데 그러기 위해서 날씨 데이터 소스 및 의류 및 스타일데이터베이스를 사용해야 하는데 그 수준으로 만들 수 없어서 scanf 함수를 이용해 입력 받기로 한다.

하지만 두께를 입력 받을 때, 옷의 두께에 따라 '두꺼움', '중간' 등의 문자열을 입력 받는 대신 두꺼우면 1을 입력 받고 중간 두께면 2를 입력 받게 함으로써 두께에 따른 의류 추천을 용의하게끔 구현해봤다.

세부 기능 (1) 사용자에게 오늘의 날씨 입력 받기

세부 기능 (2) 사용자에게 가지고 있는 옷들 입력 받기 (타입, 두께)

2) 기능 2 입력 받은 정보로 사용자에게 코디 추천

위의 입력 받은 옷들과 날씨를 바탕으로 옷의 두께에 따라 옷을 분류해서 사용자에게 옷을 추천해준다. 이 코드와 내 수준에서 옷의 상세한 사항 하나 하나씩 저장 받고 코디해주진 못하지만, 옷의 종류와 두께를 고려해서 코디해줄 수 있다.

3. 기능 구현

```
// 옷의 종류와 두께를 멤버로 갖는 구조체 정의
typedef struct
{
    char type[50];
    int thickness;
} Clothing;
```

<기능 구현을 위한 구조체>

이중포인터를 이용해서 옷의 종류와 두께를 선언했던 기존 코드에서 옷의 타입과 두께를 입력받을 멤버함수를 갖는 구조체를 선언하였다. 옷의 종류 (ex) 코트, 패딩, 바람막이 등)를 입력할 변수는 문자열로 두께는 두께에 따라 숫자로 분류하기 위해 정수형(int) 타입으로 선언하였다.

(1) recommendOutfit 함수

온도와 두께와 타입을 인수로 갖는 이 함수는 온도를 영하부터 20도 이상의 네 가지 범위로 나눠서 온도에 따른 문구와 함께 의상을 추천해주는 함수이다. 함수의 첫 부분에선 온도에 따른 날씨의 설명이 나온다. 그리고 추천 될 의상의 두께를 설정하고 함수의 매개변수 'thickness' 에 저장된다. 저장된 두께의 값에 따라 이렇게 의류를 추천해준다. 이들은 이 함수에서 출력값이다.

두께가 1이면 "두꺼운 [의상 종류]를 추천합니다."

두께가 2이면 "따뜻한 [의상 종류]를 추천합니다."

두께가 3이면 "가볍게 [의상 종류]를 추천합니다."

두께가 4이면 "시원한 [의상 종류]를 추천합니다."

이 함수에는 if 조건문이 쓰였다. 조건문과 그에 따른 출력을 용이하게 하기 위해 사용자에게 옷의 두께의 값을 숫자로 입력받게끔 했다.

```

94 // 온도에 따라 의상을 추천하는 함수
95 void recommendOutfit(int temperature, int thickness, char *type)
96 {
97     // 온도 범위에 따른 의상 추천
98     if (temperature < 0)
99     {
100         printf("오늘은 날이 매우 추우니 ");
101         // 추천 두께 설정
102         thickness = 1;
103     }
104     else if (temperature < 10)
105     {
106         printf("오늘은 날이 조금 춥습니다. ");
107         thickness = 2;
108     }
109     else if (temperature < 20)
110     {
111         printf("오늘은 날이 꽤 쌀쌀할 것 같아요. ");
112         thickness = 3;
113     }
114     else
115     {
116         printf("오늘은 따뜻한 날이 예상됩니다. ");
117         thickness = 4;
118     }
119
120     // 두께에 따라 옷을 추천
121     if (thickness == 1)
122     {
123         printf("두꺼운 %s를 추천합니다.\n", type);
124     }
125     else if (thickness == 2)
126     {
127         printf("따뜻한 %s를 추천합니다.\n", type);
128     }
129     else if (thickness == 3)
130     {
131         printf("가볍게 %s를 추천합니다.\n", type);
132     }
133     else if (thickness == 4)
134     {
135         printf("시원한 %s를 추천합니다.\n", type);
136     }

```

<코드 스크린샷>

(2)classifyAndRecommend 함수

중간 진척보고서를 적을 때 온도를 고려하지 않고 옷의 문자열을 랜덤으로 추천해주는 바람에 문제가 있다고 했는데 이를 수정했다.

이 함수는 옷을 분류하고 랜덤으로 '타입'을 선택해서 추천해주는 함수이다. 사용자에게 두께가 같은 옷을 추천 받는다면, 같은 두께들 중에서 한 가지의 옷을 추

천하기 위한 함수이다.

랜덤으로 옷을 추천하기 위해서 난수 생성을 매번 다르게 하는 함수인 `srand(time(NULL))` 을 넣어줬다.

또한 기존의 문자열을 온도를 고려하지 않고 랜덤으로 뽑는 문제를 해결하기 위해 온도를 고려한 새로운 배열을 동적으로 선언해주었다.

그리고 후보옷의 개수를 정수형 `count`로 만들고 0으로 초기화 시켜준다.

여기선 `for` 반복문안에 `if`문을 넣어줘서 온도와 두께를 모두 고려했다. 그리고 고려한 옷은 `count++`를 해줘서 늘려준다. 적합한 의상이 없을 시 나올 출력문을 만들기 위해 `count`가 0인 상황도 고려해주었다.

그리고 랜덤하게 선택된 후보 중 하나를 출력해서 실제로 옷을 추천해줄 함수인 `recommendOutfit`함수를 호출해서 옷의 두께 및 종류를 인수로 전달해준다.

마지막으로 동적으로 할당한 메모리를 해제한다.

```

void classifyAndRecommend(Clothing *clothes, int inputClothes, int temperature)
{
    srand(time(NULL)); // 난수생성을 매번 다르게 하게끔 초기화 시켜줌

    // 동적으로 후보 옷을 저장할 새로운 배열을 선언, 온도도 고려함
    Clothing **considerThickness = (Clothing **)malloc(inputClothes * sizeof(Clothing *));

    int count = 0; // 후보 옷의 개수 초기화

    // and 연산자 및 or 연산자 이용해서 온도와 두께 모두 고려하고 고려된 옷 개수 늘리기
    for (int i = 0; i < inputClothes; ++i)
    {
        if ((temperature < 0 && clothes[i].thickness == 1) ||
            (temperature < 10 && clothes[i].thickness == 2) ||
            (temperature < 20 && clothes[i].thickness == 3) ||
            (temperature >= 20 && clothes[i].thickness == 4))
        {
            considerThickness[count] = &clothes[i];
            count++;
        }
    }

    if (count == 0)
    {
        printf("해당 조건에 맞는 옷이 없습니다.\n");
        free(considerThickness); // 동적으로 할당한 메모리 해제
        return;
    }

    // 랜덤하게 선택된 후보 중 하나를 출력
    int randomIndex = rand() % count;
    printf("추천된 옷: %s\n", considerThickness[randomIndex]->type);

    // recommendOutfit 함수를 호출해서 옷의 두께 및 종류를 인수로 전달
    recommendOutfit(temperature, considerThickness[randomIndex]->thickness, considerThickness[randomIndex]->type);

    free(considerThickness); // 동적으로 할당한 메모리 해제
}

```

<코드 스크린샷>

(3) main 함수

메인함수에선 정수형 변수 temperature를 선언하고 사용자에게 직접 온도를 입력 받는다. 또한 정수형 변수 inputClothes 변수를 선언해서 가지고 있는 옷의 개수를 미리 사용자에게 전달받는다. 이때 구조체 배열에 메모리를 할당하는 malloc 함수를 이용하여 동적 할당을 함으로써 Heap 공간을 제공해준다.

InputClothes에 입력 받은 옷의 개수만큼 옷의 정보를 입력해야하므로 for 반복문에서 조건변수로 inputClothes변수를 사용, 옷의 두께는 쉽게 분류하기 위해 두께에 따라 숫자로 입력 받기로 했다. 또한 버퍼에서 개행문자를 제거하기 위해 getchar();를 사용해주었다.

그리고 사용자에게 입력받은 옷 정보와 온도를 'classifyAndRecommend'함수에

전달해서 적절한 의상을 추천받게끔 한다.

마지막으로 동적으로 할당된 clothes의 메모리를 해제하고 return 0;를 통해 프로그램 종료한다.

```
int main()
{
    int temperature;

    // 온도 정보를 입력 받는 부분 (실제로는 API를 통해 받아오겠지만, 여기에선 직접 입력 받음)
    printf("오늘의 온도는 몇 도인가요? ");
    scanf("%d", &temperature);
    // 의류 데이터베이스가 없어서 번거롭지만, 일일이 옷을 입력 받음
    int inputClothes;
    printf("가지고 있는 옷의 수를 입력하세요: ");
    scanf("%d", &inputClothes);

    // 구조체 배열 동적 할당
    Clothing *clothes = (Clothing *)malloc(inputClothes * sizeof(Clothing));

    // 각각의 옷 정보를 입력 받음.
    for (int i = 0; i < inputClothes; ++i)
    {
        printf("옷 %d 정보 입력\n", i + 1);
        printf("종류(패딩, 코트, 바람막이 등): ");
        scanf("%s", clothes[i].type);

        // 두께를 숫자로 입력받기
        printf("두께( 1(매우두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): ");
        scanf("%d", &clothes[i].thickness);

        getchar();
    }

    // 옷을 분류하고 랜덤으로 타임을 선택하여 추천하는 함수 호출
    classifyAndRecommend(clothes, inputClothes, temperature);

    free(clothes); // 메모리 해제

    return 0;
}
```

<코드 스크린샷>

4. 테스트 결과


```
문제 출력 디버그 콘솔 터미널 포트
+ Code - sources
cd "/Users/nayejun/Desktop/C202309-P/sources/" && gcc main.c -o main && "/Users/nayejun/Desktop/C202309-P/sources/"main
nayejun@nayejun-ui-MacBookPro C202309-P % cd "/Users/nayejun/Desktop/C202309-P/sources/" && gcc main.c -o main && "/Users/nayejun/Desktop/C202309-P/sources/"main
오늘의 온도는 몇 도인가요? 15
가지고 있는 옷의 수를 입력하세요: 5
옷 1 정보 입력
종류 (패딩, 코트, 바람막이 등): coat
두께 ( 1(매우 두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): 2
옷 2 정보 입력
종류 (패딩, 코트, 바람막이 등): windbreaker
두께 ( 1(매우 두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): 3
옷 3 정보 입력
종류 (패딩, 코트, 바람막이 등): nanninggu3
두께 ( 1(매우 두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): 3
옷 4 정보 입력
종류 (패딩, 코트, 바람막이 등): padding
두께 ( 1(매우 두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): 1
옷 5 정보 입력
종류 (패딩, 코트, 바람막이 등): bluezong
두께 ( 1(매우 두꺼움), 2(두꺼움), 3(중간), 4(얇음) 1~4 중 입력): 2
추천된 옷: nanninggu3
오늘은 날이 꽤 쌀쌀할 것 같아요. 가볍게 nanninggu3를 추천합니다.
nayejun@nayejun-ui-MacBookPro sources %
```

프로그램을 실행하면 첫 번째로 오늘의 온도를 물어본다. 나는 15도라고 적었다.

그러면 갖고 있는 옷의 두께와 종류를 입력 받기전에 옷의 수를 적어야 한다. 실제 API 키값과 의류데이터베이스를 가지고 프로그램을 짜면 완벽한 자동화 프로그램이 되겠지만 수준미달로 그렇게 짜진 못했다. 아무튼 옷의 개수를 입력하면 옷의 정보와 두께를 입력하게 된다.

15도라는 온도에 추천될 옷은 3번 두께이기 때문에 나는 3번 두께 옷을 두개 적고 나머지는 다른 두께 옷으로 적었다.

모든 옷의 정보를 적으면, 날씨의 수준과 그에 맞는 의류를 랜덤으로 추천해준다. 3번 두께의 옷이 바람막이와 난닝구였기 때문에 둘 중에 하나인 난닝구가 추천되었다.

5. 계획 대비 변경 사항

헤더파일을 모두 나누고 싶었는데, 구조체를 헤더파일로 나누려는 도중 너무 많은 오류가 떠서 recommendOutfit 함수만 헤더파일로 나누었다. 그리고 윈도우 환경에서 실행해보았더니 잘 되었다.

6. 느낀점

프로젝트 시작을 진도를 다 나가지 않은 채로 시작해서, 구조체나 포인터 등을

사용하지 않고 코드를 짜다가, 수업시간에 그러한 개념들을 배우고 나서 바로 나의 프로젝트에 적용시킴으로써 프로젝트의 질을 높여주는게 눈에 보였다. 사실 나의 주제와 코드들은 객관적으로 자동화 업무를 위한 코드라기엔 턱없이 부족하고 수준이 낮아서 한편으론 많이 아쉽다. 하지만 수업시간에 배운 내용들을 거의 다 코드에 반영하고, 나의 코드들을 남들에게 설명하기 위한 보고서를 적으면서 한번 더 공부가 되고 코드의 부족한 점도 보여서 자주 수정해서 여러 도움이 됐다. 이상한 완벽주의 때문에 커밋을 자주 하기보단 한번에 코드를 내는게 프로젝트의 의의에 맞지 않은 행동이었지만 다음에 이런 프로젝트를 하게 된다면 더 잘 할 자신이 생겼다.