

## 과제 1 소스코드

```
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include<string.h>
#define SIZE 5

typedef struct _student
{
    int number;
    double grade;
    char name[10];
} student;

void main()
{
    student list[SIZE];
    int i, j;
    student temp;

    for (i = 0; i < SIZE; i++)
    {
        printf("학년을 입력하시오: ");
        scanf_s("%d", &list[i].number);
        printf("학점을 입력하시오: ");
        scanf_s("%lf", &list[i].grade);
        printf("이름을 입력하시오: ");
        scanf_s("%s", list[i].name, sizeof(list[i].name));
        //그동안 해결 못하던 문제인데 scanf로 문자를 읽을때 문자열의 크기도
        지정해줘야 된다고 가르쳐 주셔서 감사합니다
    }

    // 정렬
    for (i = 0; i < SIZE - 1; i++)
    {
        for (j = 0; j < SIZE - 1 - i; j++)
        {
            if (list[j].number > list[j + 1].number || //1순위 학년
                (list[j].number == list[j + 1].number &&
                 list[j].grade > list[j + 1].grade) || //2순위 학년이 같을때 성적
                (list[j].number == list[j + 1].number &&
                 list[j].grade == list[j + 1].grade && strcmp(list[j].name, list[j + 1].name) > 0))
                //3순위 학년 성적이 같을때 문자,문자는 사전식 배열로
            {
                // 순서 배열 list[]의 순서를 배열하면 구조체 안 숫자
                두개와 문자가 동시에 바뀜
                temp = list[j];
            }
        }
    }
}
```

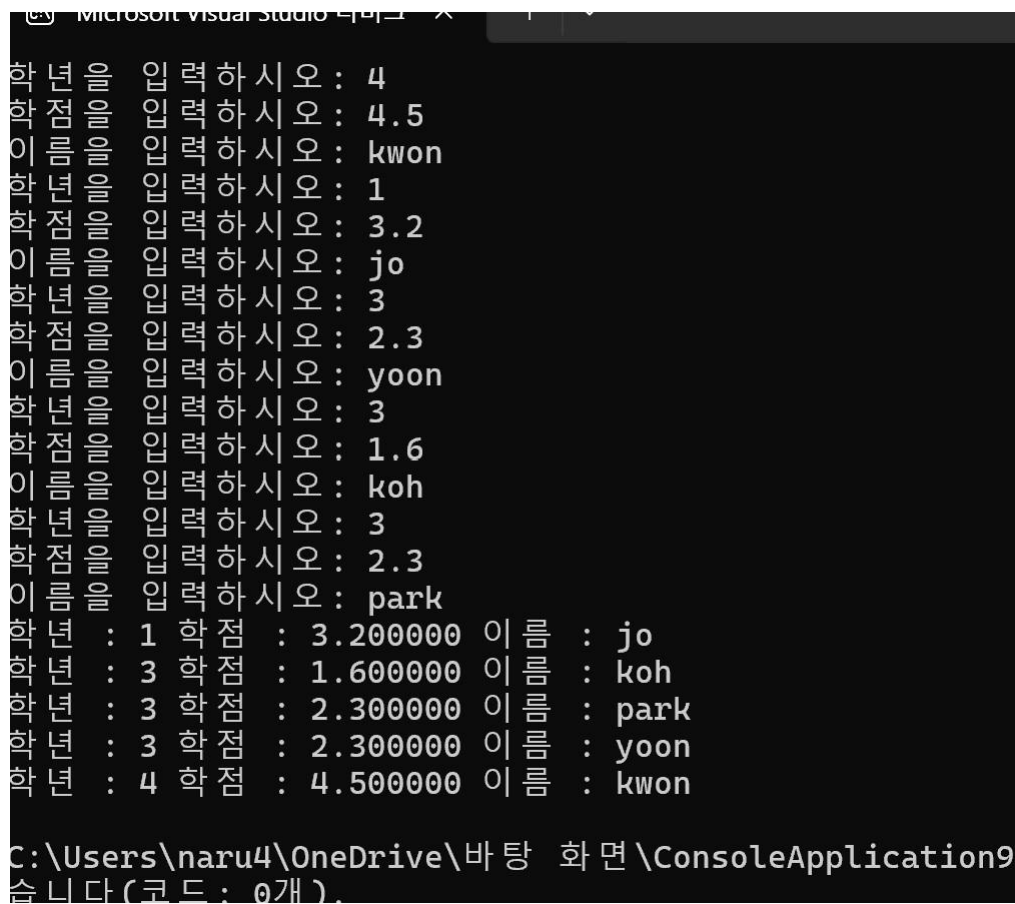
```

        list[j] = list[j + 1];
        list[j + 1] = temp;
    }
}

// 출력
for (i = 0; i < SIZE; i++)
{
    printf("학년 : %d 학점 : %f 이름 : %s\n", list[i].number,
list[i].grade, list[i].name);
}
}

```

## 실행결과



```

Microsoft Visual Studio 디버그 콘솔
학년을 입력하시오 : 4
학점을 입력하시오 : 4.5
이름을 입력하시오 : kwon
학년을 입력하시오 : 1
학점을 입력하시오 : 3.2
이름을 입력하시오 : jo
학년을 입력하시오 : 3
학점을 입력하시오 : 2.3
이름을 입력하시오 : yoon
학년을 입력하시오 : 3
학점을 입력하시오 : 1.6
이름을 입력하시오 : koh
학년을 입력하시오 : 3
학점을 입력하시오 : 2.3
이름을 입력하시오 : park
학년 : 1 학점 : 3.200000 이름 : jo
학년 : 3 학점 : 1.600000 이름 : koh
학년 : 3 학점 : 2.300000 이름 : park
학년 : 3 학점 : 2.300000 이름 : yoon
학년 : 4 학점 : 4.500000 이름 : kwon

C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication9
습니다(코드 : 0개).

```

## 과제 2 소스 코드

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

typedef struct //좌표 구조체
{
    int x;
    int y;
} Point;

double distance(Point a, Point b) //거리 계산 함수
//x,y 두개의 변수가 저장된 구조체 2개를 받으면 숫자 4개로 두 점사이의 거리를 계산
{
    return sqrt((a.x - b.x) * (a.x - b.x) + (a.y - b.y) * (a.y - b.y)); //math.h의
    sqrt=제곱근
}

void main()
{
    int num;
    Point* points;

    printf("입력 : ");
    scanf_s("%d", &num); //점의 개수 입력

    points = (Point*)malloc(sizeof(Point) * num); //동적할당

    for (int i = 0; i < num; i++)
    {
        scanf_s("%d %d", &points[i].x, &points[i].y); //좌표 입력
    }

    Point maxpoint = points[0];
    double maxdistancesum = 0; //변수 초기화

    for (int i = 0; i < num; i++) //i가 먼저 고정되서 커지는 j와 비교
    {
        double distancesum = 0;
        for (int j = 0; j < num; j++)
        {
            if (i != j) //서로 다른 두 점일때
            {
                distancesum += distance(points[i], points[j]); //i와 다른 점들의
                //거리를 합산
            }
        }
    }
}
```

```

    }
    if (distancesum > maxdistancesum)//i가 커지면서 가장 멀리 떨어져있는 좌표
찾기
    {
        maxdistancesum = distancesum;
        maxpoint = points[i];//가장 멀리 떨어져 있는 좌표 주소 저장
    }
}

printf("가장 거리가 먼 좌표는 (%d, %d)이며, 다른 좌표의 거리 총합은
약 %lf입니다.\n", maxpoint.x, maxpoint.y, maxdistancesum);//출력
free(points);//동적 할당 해제
}

```

실행 결과

```

입력 : 5
0 0
1 1
2 3
2 1
7 7
가장 거리가 먼 좌표는 (7, 7)이며, 다른 좌표의 거리 총합은 약 32.598150입니다.

C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication11\x64\Debug\ConsoleApplication11.
었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 종
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요 ...|

입력 : 3
1 2
7 8
-1 2
가장 거리가 먼 좌표는 (7, 8)이며, 다른 좌표의 거리 총합은 약 18.485281입니다.

C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication11\x64\Debug\ConsoleApplication
었습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요 ...|

```

### 과제 3 소스코드

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define _CRT_SECURE_NO_WARNINGS

// 물품 가격 구조체
typedef struct
{
    char name[50];
    int price;
} Item;

int main()
{
    int num;
    printf("입력 : ");
    scanf_s("%d", &num);

    // 동적 할당
    Item* itemlist = (Item*)malloc(sizeof(Item)*num);

    // 물품과 금액 입력 받기
    for (int i = 0; i < num; i++)
    {
        scanf_s("%s", itemlist[i].name, sizeof(itemlist[i].name));
        scanf_s("%d", &itemlist[i].price);
    }

    // 합계를 구할 물품 이름 입력 받기
    char check[20];
    scanf_s("%s", check, sizeof(check));

    // 합계 계산
    int sum = 0; //변수 초기화
    for (int i = 0; i < num; i++)
    {
        if (strcmp(itemlist[i].name, check) == 0) //합계를 구할 물품의 이름이라면
        {
            sum += itemlist[i].price; //합계 계산
        }
    }

    // 결과 출력
    printf("출력 : %d\n", sum); //금액 입력이 없던 이름이라면 sum=0
```

```

    // 동적 할당 메모리 해제
    free(itemlist);
    return 0;
}

```

실행결과

```

입력 : 3
apple
2000
coffee
1000
apple
1500
apple
출력 : 3500

```

C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication13\x64\Debug\ConsoleApplication13.exe  
 었습니다(코드 : 0개).  
 디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] -> [콘솔] 탭에서  
 하도록 설정합니다.  
 이 창을 닫으려면 아무 키나 누르세요...

```

입력 : 5
apple
2000
coffe
1000
apple
1500
tea
300
apple
200
orange
출력 : 0

```

C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication13\x64\Debug\ConsoleApplication13.exe  
 었습니다(코드 : 0개).  
 디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] -> [콘솔] 탭에서  
 하도록 설정합니다.  
 이 창을 닫으려면 아무 키나 누르세요...

## 과제 4 소스코드

```
#include <stdio.h>

typedef struct _date
{
    int year, month, day, hour, min, sec;
} date;

// 윤년인지 확인
int leapyear(int year)
{
    return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);
}

// 월별 일수를 반환하는 함수
int daysum(int year, int month)
{
    if (month == 2)
    {
        if (leapyear(year) == 1) //윤년이라면
        {
            return 28; //28일
        }
        else
            return 29; //아니면 29일
    }
    else if (month == 4 || month == 6 || month == 9 || month == 11) //30일 있는 달
    {
        return 30;
    }
    else
    {
        return 31; //나머지는 다 31일
    }
}

void main()
{
    int yearcount, monthcount, daycount, hourcount, mincount, seccount,
    resultcount;
    int printhour, printmin, printsec;
    date list[2]; //년,달,월,일,시,분,초가 포함되어있는 구조체 2개
    printf("입력 : %n");
    for (int i = 0; i < 2; i++)
    {
```

```

scanf_s("%d %d %d %d %d %d", &list[i].year, &list[i].month, &list[i].day,
&list[i].hour, &list[i].min, &list[i].sec);
}

// 년도 차이 계산
yearcount = (list[1].year - list[0].year) * 365 * 24 * 60 * 60;

// 월 차이 계산
monthcount = 0;
if (list[1].year == list[0].year) //연도 차이가 아닐때
{
    if (list[1].month > list[0].month)
    {
        for (int i = list[0].month; i < list[1].month; i++) //i가 달 차이
수만큼 반복
        {
            monthcount += daysum(list[0].year, i) * 24 * 60 * 60; //달마다 일수
총합
        }
    }
}
else //연도 차이가 날때
{
    for (int i = list[0].month; i <= 12; i++) //다음 년까지 남은 달의 수
    {
        monthcount += daysum(list[0].year, i) * 24 * 60 * 60;
    }
    for (int i = 1; i < list[1].month; i++) //다음 년부터 지나갈 달의 수
    {
        monthcount += daysum(list[1].year, i) * 24 * 60 * 60;
    }
}
//만약 나중 시간이 전 시간 보다 적은 계산 값이 있다면 -로 계산됨
// 일 차이 계산
daycount = (list[1].day - list[0].day) * 24 * 60 * 60;

// 시간, 분, 초 차이 계산
hourcount = (list[1].hour - list[0].hour) * 60 * 60;
mincount = (list[1].min - list[0].min) * 60;
seccount = list[1].sec - list[0].sec;

// 총 초 계산
resultcount = yearcount + monthcount + daycount + hourcount + mincount +
seccount;

// 초를 시, 분, 초로 변환
printhour = (resultcount - (resultcount % 3600)) / 3600;

```



```

    printsec = (resultcount - (printhour * 3600)) % 60;
    printmin = (resultcount - (printhour * 3600) - printsec) / 60;

    // 출력
    printf("%d시 %d분 %d초", printhour, printmin, printsec);
}

```

윤년이 아닌 년중 예시의 숫자를 입력하면 똑같이 나옴

```

입력 :
2025 01 01 10 20 30
2025 06 18 12 05 10
4057시 44분 40초
C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication12\x64\Debug\ConsoleApplication12.exe
있습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지될 때 콘솔을 닫음]을
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...|

```

예시의 숫자는 윤년을 고려 하지 않은것으로 생각

```

입력 :
2024 01 01 10 20 30
2024 06 18 12 05 10
4033시 44분 40초
C:\Users\naru4\OneDrive\바탕 화면\ConsoleApplication12\x64\Debug\ConsoleApplication12.exe
있습니다(코드: 0개).
디버깅이 중지될 때 콘솔을 자동으로 닫으려면 [도구] -> [옵션] -> [디버깅] > [디버깅이 중지될 때 콘솔을 닫음]을
하도록 설정합니다.
이 창을 닫으려면 아무 키나 누르세요...|

```