

컴퓨터정보공학부 2021202058 송채영

문제 1 (소요시간: 20분)

소스코드

```
#include <stdio.h>
max_ptr(int m[], int size, int** pmax); //배열의 큰값을 가르키게 하는 함수

int main()
{
    int m[5] = { 10, 20, 30, 40, 50 }; //배열에 임의의 초기값을 줌
    int* pmax; //가장 큰 수를 가르킬 포인터 변수 pmax 선언

    max_ptr(m, 5, &pmax); //pmax의 주소 &pmax는 이중포인터

    printf("가장 큰 값은 %d", *pmax); //가장 큰 값 출력
    return 0;
}

max_ptr(int m[], int size, int** pmax) //배열의 큰값을 가르키게 하는 함수
{
    int max;
    max = m[0];
    for (size = 0; size < 5; size++)
    {
        if (max < m[size])
        {
            max = m[size];
            *pmax = &m[size];
        }
    }
}
```

결과화면

```
가장 큰 값은 50
C:\Users\#82104\source\repos\Project6\#x64\Debug\Project6.exe(프로세스 7556개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요....
```

문제 2 (소요시간: 20분)

소스코드

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
void get(double, int*, double* ); //정수부와 실수부로 나누는 함수

int main()
{
    double value = 0;
    int integer; //정수
```

```

    double real; //실수
    printf("실수를 입력하세요 : ");
    scanf("%lf", &value); //실수 입력

    get(value, &integer, &real); //정수부와 실수부로 나누는 함수에 값 넣어줌

    printf("정수부 : %d\n", integer);
    printf("실수부 : %f\n", real);
    return 0;
}

void get(double value, int* integer, double* real) //정수부와 실수부로 나누는 함수
{
    *integer = (int)value;
    *real = value - *integer;
}

```

결과화면



```

Microsoft Visual Studio 디버그 콘솔
실수를 입력하세요 : 3.5
정수부 : 3
실수부 : 0.500000
C:\Users\82104\source\repos\Project6\Debug\Project6.exe(프로세스 21440개)이(가) 종료되었습니다(코드: 0개)
이 창을 닫으려면 아무 키나 누르세요...

```

고찰

scanf("%d", &value);로 써주어 실수부와 정수부에 0과 0.000000의 값만 떴었다. double형의 실수로 입력하기 위해서는 lf를 써주어야 하는데, 이 점을 까먹고 있어 계속된 오류가 났었다.

문제 3 (소요시간: 2시간)

소스코드

```

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

int main(int argc, char* argv[]) //char* argv = char **argv, argc는 메인 함수에 전달되는 정보의 갯수, argv는 문자열의 배열(메인함수에 전달되는 정보)
{
    char msg[10][50]; //문자열을 저장할 배열

    printf("argc : %d\n", argc);

    printf("명령어 라인 문자열이 저장된 msg 2차원 문자배열\n");
    int i;
    for (i = 0; i < argc; i++)
    {
        strcpy(msg[i], argv[i + 1]); //argv[0]에 프로그램의 실행경로가 들어감
    }
}

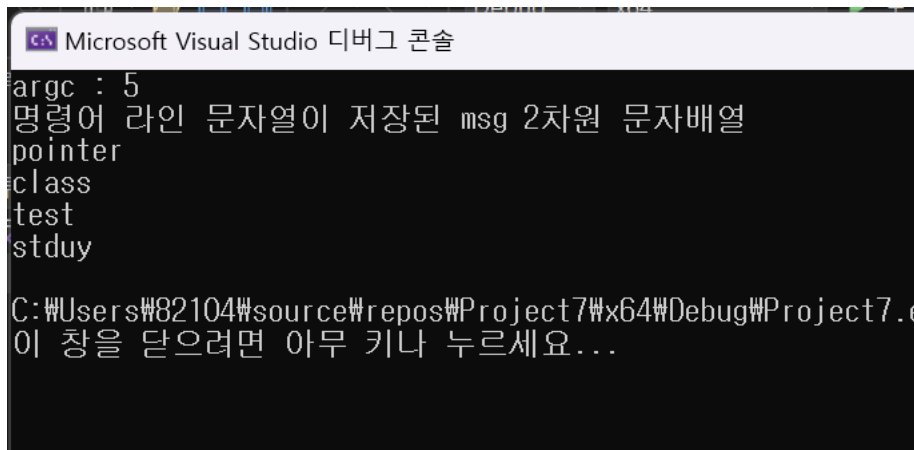
```

```

        printf("%s\n", msg[i]); //문자열 출력
    }
    return 0;
}

```

결과화면



```

Microsoft Visual Studio 디버그 콘솔
argc : 5
명령어 라인 문자열이 저장된 msg 2차원 문자배열
pointer
class
test
stduy

C:\Users\82104\source\repos\Project7\Debug\Project7.exe
이 창을 닫으려면 아무 키나 누르세요...

```

```

C:\Users\82104\source\repos\Project7\Debug>Project7.exe pointer class test study
argc : 5
명령어 라인 문자열이 저장된 msg 2차원 문자배열
pointer
class
test
study

```

고찰

첫 번째 사진은 visual studio 속성페이지에서 명령 인수에 직접 입력해 출력한 경우이고, 두 번째 사진은 cmd(명령 프롬프트)를 이용한 출력에서 시간이 오래 걸렸다. Cd를 통해서 폴더에 접근하는 것은 알고 있었는데, 출력하는 부분에서 시간이 오래 걸렸던 것 같다.