

컴퓨터 프로그래밍 2
-hw01-

학번 : 201602038
이름 : 이 미 진

1. 함수 설명

1) void AppIO_out_msg_startSolvingLinearEquation() : <<<일차방정식 풀이>>> 메시지를 출력하는 함수

2) void AppIO_out_msg_endSolvingLinearEquation() : <<<일차방정식 풀이를 종료합니다>>> 메시지를 출력하는 함수

3) void AppIO_in_solvingIsRequested() : 방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 입력받고 입력받은 값이 y일 때 풀이가 시작된다는 내용의 함수

4) void AppIO_in_linearEquation (float* p_c0, float* p_c1) : 1차항의 계수와 상수항의 계수를 입력받는 함수

5) void AppIO_out_linearEquation(float c0, float c1) : 일차방정식을 보여주는 함수

6) void AppIO_out_root(float root) : 방정식의 해를 출력해주는 함수

7) void AppIO_out_msg_firstOrderTermCoefficientIsZero() : 일차항의 계수가 0일 때 “일차항의 계수가 0입니다.” 라는 메시지를 출력하는 함수

8) float LinearEquation_solve (float c0, float c1) : 주어진 계수를 갖는 일차방정식의 해를 구하는 함수

2. 전체 코드

1) main.c

```
//
//  main.c
//  cp2
//
//  Created by stu2017s10 on 2017. 3. 7..
//  Copyright © 2017년 stu2017s10. All rights reserved.
//

#include <stdio.h>
#include<stdlib.h>
#include<math.h>    // 수학 함수 사용을 위한 math.h 라이브러리
#include "Common.h" // 매크로 헤더
#include "AppIO.h"  // AppIO 관련 함수 헤더

float LinearEquation_solve (float c0, float c1) {    // 주어진 계수를 갖
는 일차방정식의 해를 구하는 함수
    if(FloatValueIsZero(c1)) {    // 코드의 안전성을 위해 매크로를 통해 검사
        if(c1*c0 >= 0.0) {    // 두 계수의 부호가 같을 때
            return -MAXFLOAT;    // return 값은 -MAXFLOAT;
        }
        else {    //두 계수의 부호가 다를 때
            return +MAXFLOAT;    // return 값은 +MAXFLOAT;
        }
    }
    else {
        return (-c0)/c1;
    }
}

int main( int argc, const char * argv[])
{
    Boolean solvingIsRequested;
    float c0,c1;    //c1 : 1차항의 계수 , c0 : 상수항의 계수
    float root;

    AppIO_out_msg_startSolvingLinearEquation(); // <<<일차방정식 풀이>>>
메시지를 출력

    solvingIsRequested = AppIO_in_solvingIsRequested(); // 방정식을 풀
려면 y, 풀이를 종료하려면 아무 문자나 입력받고 입력받은 값이 y일 때 풀이가 시작된다는 내용의
함수

    while(solvingIsRequested) { //반복해야 할 일
        AppIO_in_linearEquation(&c0,&c1);    //일차방정식의 계수를 입력받는
다.
```

```

AppIO_out_linearEquation(c1,c0);    //입력 받은 일차방정식을 보여준
다.

    if(FloatValueIsZero(c1)) {    //일차항의 계수가 0이면
        AppIO_out_msg_firstOrderTermCoefficientIsZero();    //
일차항의 계수가 0이라는 메시지를 출력
    }
    else {
        root = LinearEquation_solve(c0,c1); //일차방정식을 푼다.
        AppIO_out_root(root);    //일차방정식의 해를 출력
    }
    solvingIsRequested = AppIO_in_solvingIsRequested(); // 방정식
을 풀려면 y, 풀이를 종료하려면 아무 문자나 입력받고 입력받은 값이 y일 때 풀이가 시작된다는 내
용의 함수
    }
    AppIO_out_msg_endSolvingLinearEquation();    // <<<일차방정식 풀이를
종료합니다>>> 메시지를 출력

    return 0;
}

```

2) Common.h

```

//
// Common.h
// cp2
//
// Created by stu2017s10 on 2017. 3. 8..
// Copyright © 2017년 stu2017s10. All rights reserved.
//

#ifndef C1_LinearEquation_Common_h
#define C1_LinearEquation_Common_h

#define EPSILON 1.0E-6 // 상수에 이름을 부여하기 위한 매크로 선언
#define FloatValueIsZero(NUMBER) fabsf(NUMBER)<EPSILON // 반복 사용되는
코드를 위한 매크로 선언

typedef enum {FALSE,TRUE} Boolean; // Boolean 타입을 define 해주고
FALSE, TRUE 값을 가짐. FALSE와 TRUE의 순서는 바뀌면 안됨.

#endif /* Common_h */

```

3) AppIO.c

```
//
// AppIO.c
// cp2
//
// Created by stu2017s10 on 2017. 3. 8..
// Copyright © 2017년 stu2017s10. All rights reserved.
//

// AppIO 관련 함수 구현
#include "AppIO.h" // AppIO 헤더 파일 include

void AppIO_out_msg_startSolvingLinearEquation() { // <<<일차방정식 풀
이>>> 메시지를 출력
    printf("<<<일차방정식 풀이>>>\n");
}
void AppIO_out_msg_endSolvingLinearEquation() { // <<<일차방정식 풀이를
종료합니다>>> 메시지를 출력
    printf("<<<일차방정식 풀이를 종료합니다>>>\n");
}
void AppIO_out_linearEquation(float c0, float c1) { // 일차방정식을 보여주
는 함수
    printf(">주어진 방정식: (%f)x+(%f) = 0\n" ,c1,c0);
}

void AppIO_out_root(float root) { // 방정식의 해를 출력
    printf(">방정식의 해는 다음과 같습니다 : \n");
    printf("x = %f\n",root);
}
void AppIO_out_msg_firstOrderTermCoefficientIsZero() { // 일차항의 계
수가 0일 때, 일차항의 계수가 0이므로 방정식을 풀 수 없다는 메시지를 출력
    printf(">일차항의 계수가 0이므로 방정식을 풀 수 없습니다.\n");
}

Boolean AppIO_in_solvingIsRequested() { // 방정식을 풀려면 y, 풀이를 종료하려
면 아무 문자나 입력하라는 메시지를 출력하고 사용자에게 입력을 받아 입력받은 값이 y일 때 풀이가
시작되고 그 외의 문자일 땐 종료한다.
    char answer;
    printf("방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.");
    fpurge(stdin); // 이전에 입력되어 임시 보관중인 모든 입력 값을 없앤다.
    answer = getchar(); // 새로 입력하는 값을 받는다.
    return (answer == 'y'); // 입력받은 값이 y이면 TRUE,그 외의 값은 FALSE를
return
}

void AppIO_in_linearEquation (float* p_c0, float* p_c1) { // 1차항의
계수와 상수항의 계수를 입력받는 함수
```

```

printf("1차항의 계수를 입력하시오: ");
scanf("%f",p_c1);
printf("상수항의 계수를 입력하시오: ");
scanf("%f",p_c0);
}

```

4) AppIO.h

```

//
// AppIO.h
// cp2
//
// Created by stu2017s10 on 2017. 3. 8..
// Copyright © 2017년 stu2017s10. All rights reserved.
//

#ifndef AppIO_h
#define AppIO_h

#include <stdio.h>
#include "Common.h" // Boolean 타입을 위해 선언

void AppIO_out_msg_startSolvingLinearEquation(); // <<<일차방정식 풀
이>>> 메시지를 출력
void AppIO_out_msg_endSolvingLinearEquation(); // <<<일차방정식 풀이를
종료합니다>>> 메시지를 출력
void AppIO_out_linearEquation(float c0, float c1); // 일차방정식을 보여주
는 함수
void AppIO_out_root(float root); // 방정식의 해를 출력해주는 함수
void AppIO_out_msg_firstOrderTermCoefficientIsZero(); // 일차항의 계
수가 0일 때 일차항의 계수가 0이므로 방정식을 풀 수 없다는 메시지를 출력
Boolean AppIO_in_solvingIsRequested(); // 방정식을 풀려면 y, 풀이를 종료하려면
아무 문자나 입력하라는 메시지를 출력하고 사용자에게 입력을 받아 입력받은 값이 y일 때 풀이가 시
작되고 그 외의 문자일 땐 종료한다.
void AppIO_in_linearEquation (float* p_c0, float* p_c1); // // 1차항의
계수와 상수항의 계수를 입력받는 함수

#endif /* AppIO_h */

```

3. 종합 설명

void AppIO_out_msg_startSolvingLinearEquation() 함수를 통해 “<<<일차방정식 풀이>>>” 라는 시작 메시지를 출력하고,
Boolean AppIO_in_solvingIsRequested() 함수를 호출해 “방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.” 라는 메시지를 출력한다.
그 다음 y가 입력되면 일차방정식 풀이를 위한 입력을 받고, 그 외의 값이 입력되면 프로그램을 종료한다.
y가 입력되면 void AppIO_in_linearEquation (float* p_c0, float* p_c1) 함수로 일차방정식을 위한 1차항의 계수와 상수항의 계수를 입력받는다.
입력을 받으면 AppIO_out_linearEquation(c1,c0) 함수로 입력 받는 일차방정식을 보여주고
만약 일차항의 계수가 0이면 AppIO_out_msg_firstOrderTermCoefficientIsZero() 함수로 일차항의 계수가 0이므로 일차방정식을 풀 수 없다는 메시지를 출력해준다.
일차항의 계수가 0이 아니라면 float LinearEquation_solve (float c0, float c1) 함수를 통해 일차방정식을 푼다.
이때, 코드의 안전성을 검사하기 위해 Common.h 라는 헤더 파일로 상수에 이름을 부여하고 반복 사용되는 코드를 위한 매크로를 선언하여 사용한다.
void AppIO_out_root(float root) 함수로 계산된 방정식의 해를 보여준다.
해를 보여주고 다시 Boolean AppIO_in_solvingIsRequested() 함수를 호출해 “방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.” 라는 메시지를 출력한다.
y가 입력되면 위의 과정을 반복하고 그 외의 값이 입력되면 프로그램을 종료한다.
결과적으로 1차항의 계수와 상수항의 계수를 입력받아 그 값으로 일차방정식을 하는 프로그램으로, 일차방정식 해 x에 관한 계산 결과 값이 출력된다.

4. 실행 결과

```
<<<일차방정식 풀이>>>
방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.y
1차항의 계수를 입력하십시오: 2.0
상수항의 계수를 입력하십시오: 3.0
>주어진 방정식: (3.000000)x+(2.000000) = 0
>방정식의 해는 다음과 같습니다 :
x = -1.500000
방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.y
1차항의 계수를 입력하십시오: 0.0
상수항의 계수를 입력하십시오: 1.0
>주어진 방정식: (1.000000)x+(0.000000) = 0
>일차항의 계수가 0이므로 방정식을 풀 수 없습니다.
방정식을 풀려면 y, 풀이를 종료하려면 아무 문자나 치시오.
```