

2022년 1학기

프로그래밍과 문제해결

Assignment #1

담당교수: 윤은영

학번: 20220100

학과: 무은재학부

이름: 박기현

POVIS ID: kihyun

명예서약(Honor Code)

“나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.”

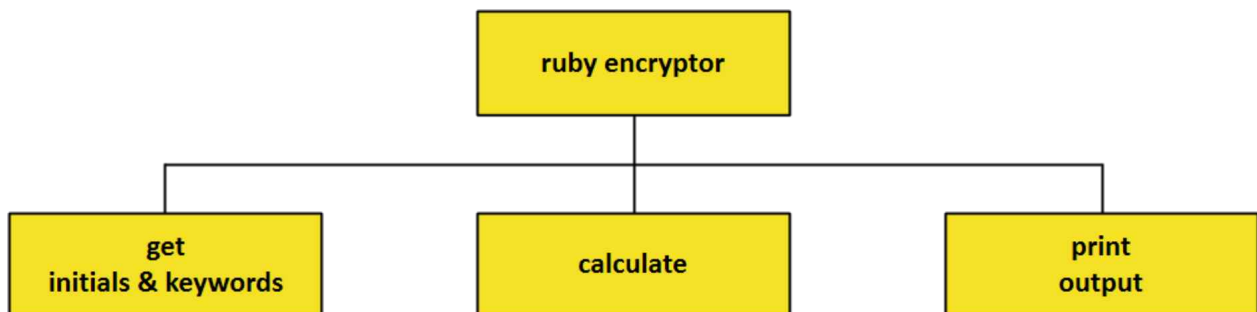
Problem 1. 루비 암호를 이용해 자신의 소문자 이니셜 암호화하기

1. 문제의 개요

본 프로그램을 간략히 설명하면 다음과 같다.

- 본인의 이니셜을 알파벳 소문자로 입력 받는다.
- 암호화에 사용될 소문자 세 자리의 키워드를 입력 받는다.
- 입력 받은 소문자 이니셜 각각을 세 자리의 키워드를 이용해 암호화한다.
- 암호화한 소문자 이니셜을 대문자로 변환하여 출력한다.

이 때 사용되는 구상 가능한 구조 차트(structure chart)는 아래와 같이 표현될 수 있다.



2. 알고리즘

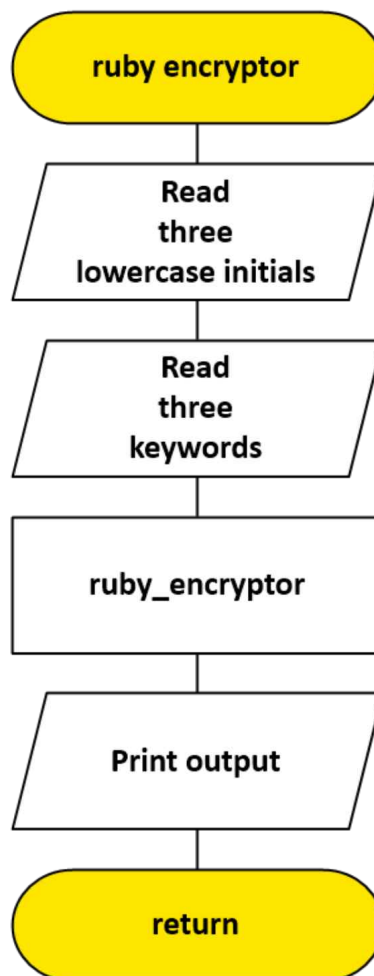
본 프로그램 작성을 위한 알고리즘을 Pseudo 코드 형태로 나타내면 다음과 같다.

Pseudo-algorithm for encrypting lowercase initials using Ruby code

// 프로그램에 필요한 변수들은 미리 선언해놓은 것으로 가정한다.

- 1 include an essential header <stdio.h>
 - 2 prompt user and read three lowercase initials
 - 3 prompt user and read three keywords
 - 4 calculate the output using ruby code
 - 5 $\text{ruby_encryptor} = ((\text{input} - 19) + (\text{key} - 97)) \% 26 + 65$
 - 6 return output
 - 7 print output
-

위의 의사 알고리즘을 flow chart를 통해 표현하면 아래와 같다.



3. 프로그램 구조 및 설명

a) 소문자 이니셜 및 키워드 입력

- 프로그램을 실행하면 소문자 이니셜 세 자리를 입력할 것을 요청한다. 사용자가 공백을 구분하여 이니셜을 입력하면 scanf를 통해 문자형 변수 input1, input2, input3에 각각 저장된다.
- 이 후, 위와 유사하게 키워드 세 자리를 입력할 것을 요청하며, 같은 방식으로 scanf를 통해 문자형 변수 key1, key2, key3에 각각 저장된다.

b) ASCII 코드를 이용한 루비 암호화

- 입력을 마친 후 각 변수에 저장된 값을 통해 출력값을 계산한다. 이 때, 루비 암호와 ASCII 코드를 이용하여 계산한다.
- 'a'의 ASCII 코드 값을 산술 연산자 %를 이용하여 알파벳 개수인 26으로 나눈 나머지 값인 19를 입력값에 빼주어 'a' = 0, 'b' = 1, ..., 'z' = 25, 즉 오름차순이 될 수 있도록 계산한다.
- 키워드의 ASCII 코드 값에 'a'의 ASCII 코드 값을 빼주어 몇 칸을 이동해야 하는지 계산한다.
- 앞서 구한 두 값을 더해 산술 연산자 %를 이용하여 나머지를 구하고, 구한 값의 ASCII 코드 값에 'A'의 ASCII 코드 값인 65를 더해주어 출력값을 구한다.

c) ASCII 코드를 통한 루비 암호화된 대문자 이니셜 출력

- 아래 소스 코드를 컴파일하고 실행하면, 각 영문에 해당하는 ASCII 코드를 알 수 있다.

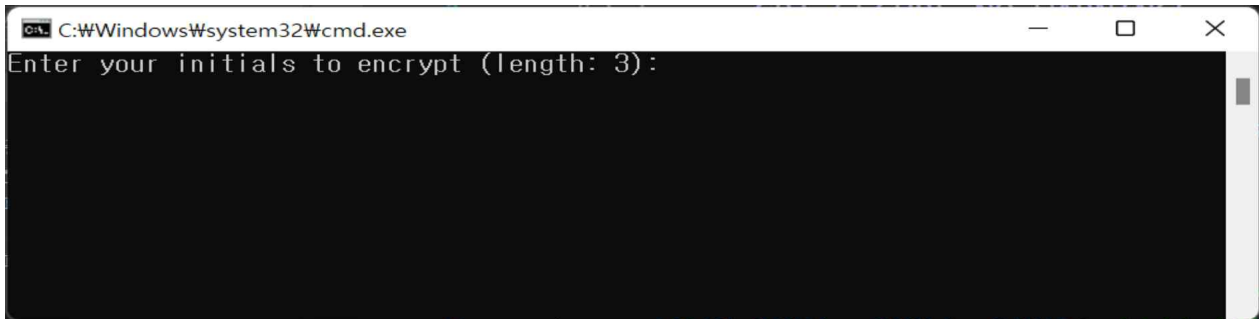
Source-code for ASCII code of input

```
#include <stdio.h>
int main(){
    char initial;
    printf("Input a character: ");
    scanf("%c", &initial);
    printf("\nASCII code of your input: %d\n", initial);
    return 0;
}
```

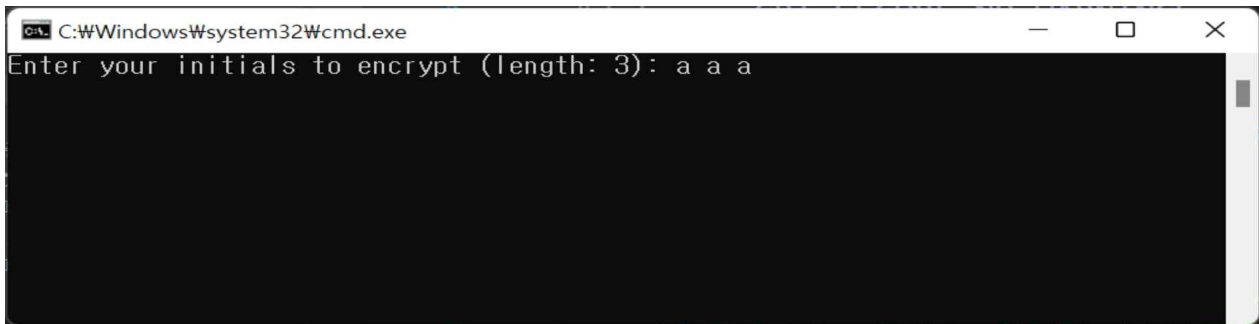
- 또는 ASCII code table을 참조하여 각 문자에 해당하는 ASCII 코드를 찾을 수 있다.
- printf("%c%c%c ...", ASCII_1, ASCII_2, ...); 포맷을 활용하여 출력한다.

4. 프로그램 실험방법 및 예제

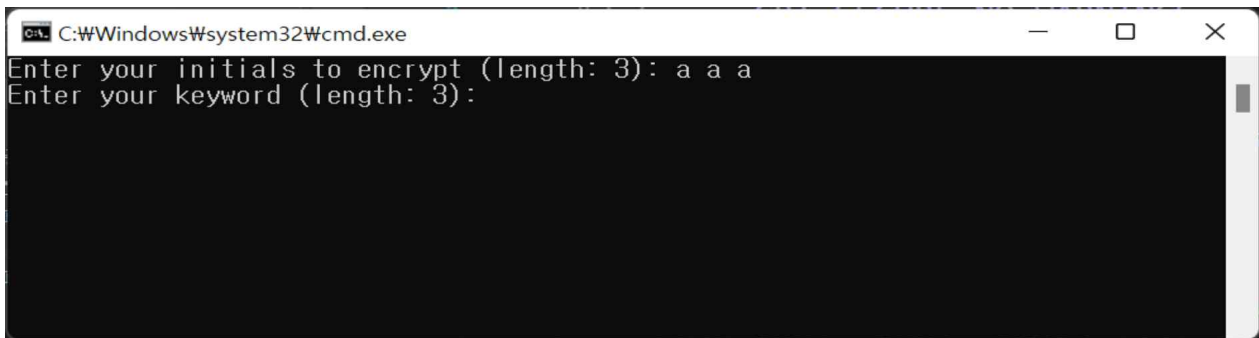
(첫 번째 예시)



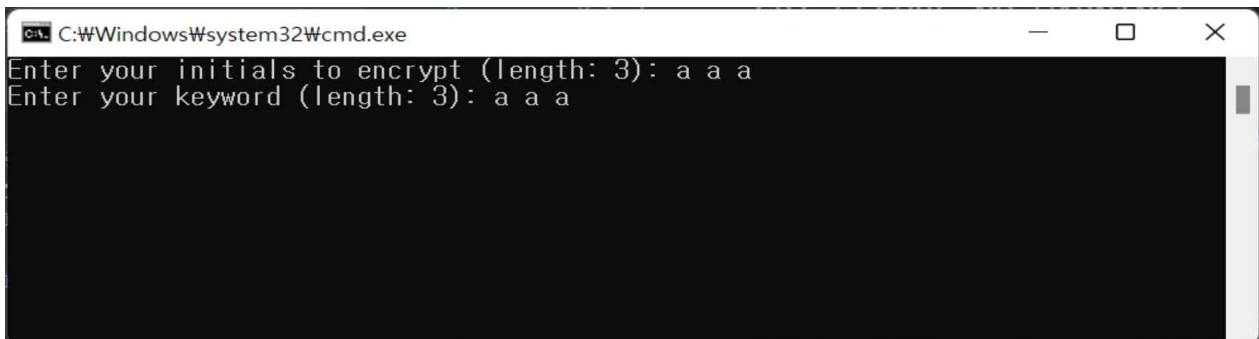
프로그램을 실행하면 소문자 이니셜 입력을 요구한다.



공백을 기준으로 세 자리 소문자 이니셜을 입력한다.



다음으로 키워드 세자리 입력을 요구한다.



공백을 기준으로 키워드 세자리를 입력한다.

```
C:\Windows\system32\cmd.exe
Enter your initials to encrypt (length: 3): a a a
Enter your keyword (length: 3): a a a
AAA
계속하려면 아무 키나 누르십시오 . . .
```

루비 암호화한 대문자 이니셜을 출력한다.

(두 번째 예시)

```
C:\Windows\system32\cmd.exe
Enter your initials to encrypt (length: 3): h j e
Enter your keyword (length: 3): y e s
FNW
계속하려면 아무 키나 누르십시오 . . .
```

5. 토론

- 루비 암호화하는 과정에서 'z' 다음으로 'a'가 나올 수 있도록 하는 과정이 쉽지 않았지만, 산술 연산자 %와 ASCII 코드 값을 통해 해결할 수 있었다.
- getchar(); 에 대해서 아직 이해를 하지 못 한 부분이 많다.

6. 결론

- 본 과제는 printf와 scanf, 사용자 정의 함수, 산술 연산자 %, ASCII 코드를 정확히 이해하고 있어야 하는 문제였다. 문제를 해결하는 과정에서 문자 및 정수 변수의 관계, 입력버퍼 제거, 사용자 정의 함수 등을 확실히 학습할 수 있게 되었다.

7. 개선방향

- 앞으로 학습하게 될 사용자 정의 함수, 포인터, 반복문/조건문 등을 통해 프로그램이 더욱 원활히 작동될 수 있도록 할 것이다.

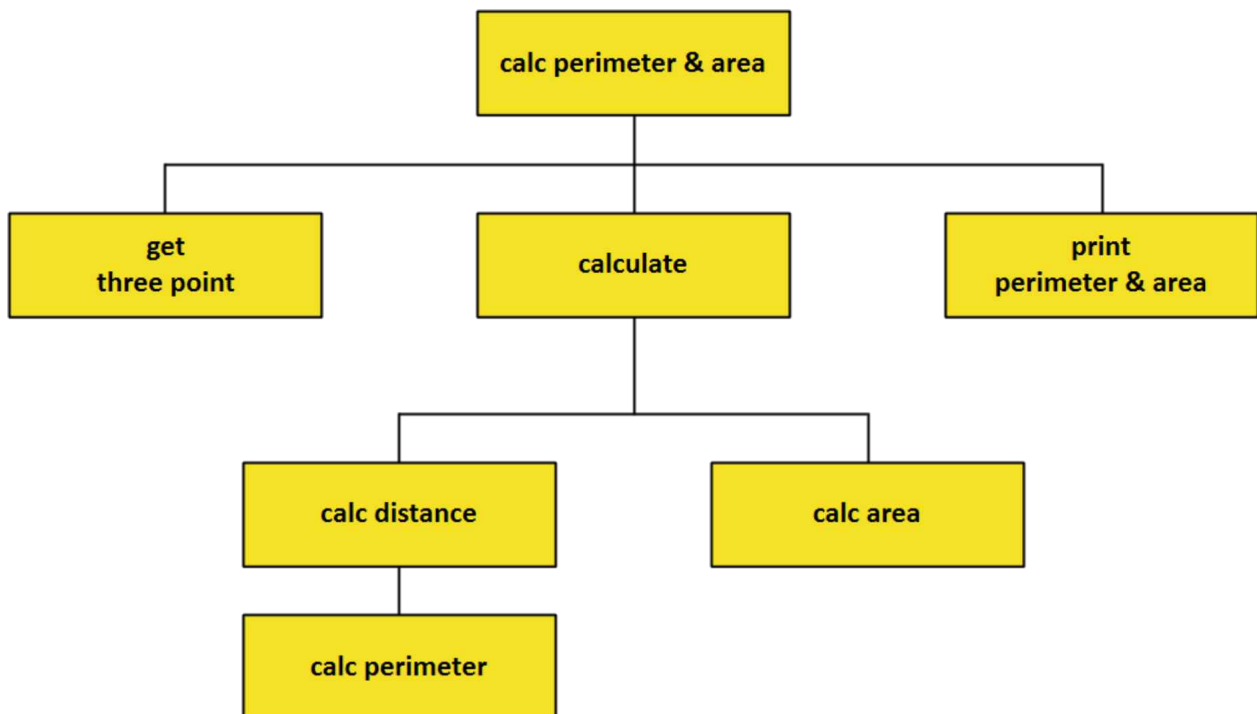
Problem 2. 좌표 평면 위 세 점의 좌표를 알 때, 삼각형 둘레의 길이와 면적 구하기

1. 문제의 개요

본 프로그램을 간략히 설명하면 다음과 같다.

- 한 직선 위에 있지 않은 서로 다른 3개의 좌표를 정수 타입으로 입력받는다.
- 삼각형 둘레의 길이를 계산한다.
- 삼각형의 넓이를 계산한다.
- 소수점 둘째 자리까지 출력한다.

이 때 사용되는 구상 가능한 구조 차트(structure chart)는 아래와 같이 표현될 수 있다.



2. 알고리즘

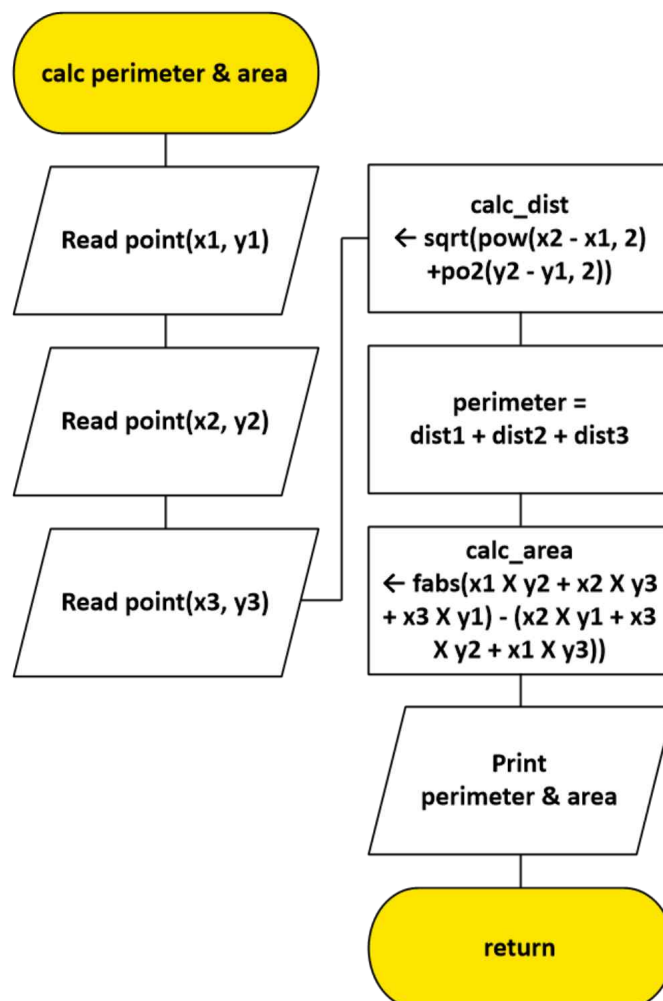
본 프로그램 작성을 위한 알고리즘을 Pseudo 코드 형태로 나타내면 다음과 같다.

Pseudo-algorithm for calculating the perimeter and the area of a triangle

// 프로그램에 필요한 변수들은 미리 선언해놓은 것으로 가정한다.

```
1 include an essential header <stdio.h> & <math.h>
2 prompt user and read first point(x1,y1)
3 prompt user and read second point(x2,y2)
4 prompt user and read third point(x3,y3)
5     calc_dist = sqrt(pow(x2 - x1, 2) + pow(y2 - y1, 2))
        calc_area = fabs((x1 * y2 + x2 * y3 + x3 * y1) - (x2 * y1 + x3 * y2 + x1 * y3)) / 2
6 return perimeter and area
7 print perimeter and area
```

위의 의사 알고리즘을 flow chart를 통해 표현하면 아래와 같다.



3. 프로그램 구조 및 설명

a) 세 점 좌표 입력

- 프로그램을 실행하면 세 개의 좌표를 입력할 것을 요청한다. 사용자가 공백을 구분하여 순서대로 좌표값을 입력하면 scanf를 통해 정수형 변수 px1, py1, px2, py2, px3, py3에 각각 저장된다.


b) 삼각형 둘레의 길이 계산

- 입력을 마친 후 각 변수에 저장된 값을 통해 삼각형 둘레의 길이를 계산한다.
- calc_dist 함수를 이용해 두 점의 길이를 각각 세 번 구한 후, 세 번의 길이를 더해 삼각형 둘레의 길이를 구한다.

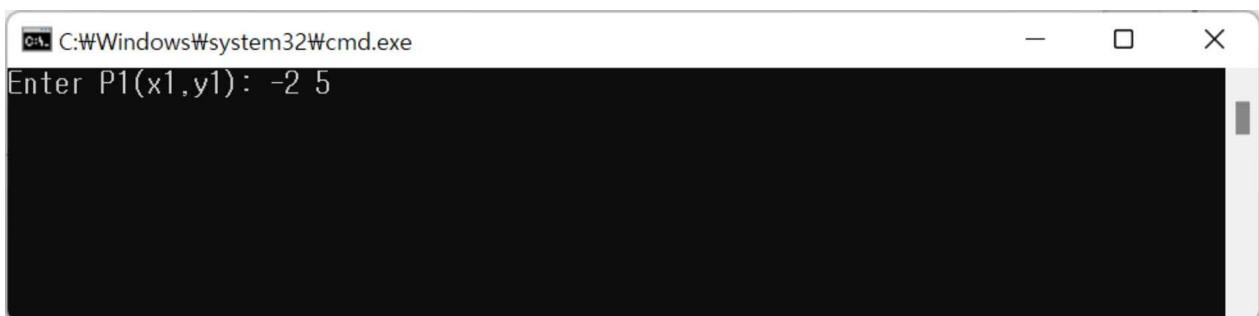
c) 삼각형 넓이 계산

- 입력을 마친 후 각 변수에 저장된 값을 통해 삼각형 넓이를 계산한다.
- calc_area 함수를 이용해 삼각형 넓이를 구한다.

4. 프로그램 실험방법 및 예제



프로그램을 실행하면 첫 번째 좌표 입력을 요구한다.



공백을 기준으로 좌표값을 입력한다.

```
C:\Windows\system32\cmd.exe
Enter P1(x1,y1): -2 5
Enter P2(x2,y2):
```

두 번째 좌표 입력을 요구한다.

```
C:\Windows\system32\cmd.exe
Enter P1(x1,y1): -2 5
Enter P2(x2,y2): 1 3
```

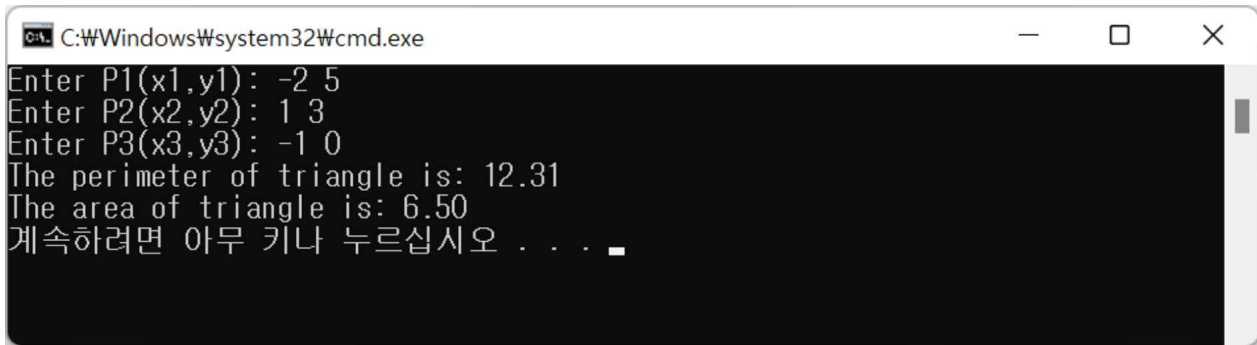
공백을 기준으로 좌표값을 입력한다.

```
C:\Windows\system32\cmd.exe
Enter P1(x1,y1): -2 5
Enter P2(x2,y2): 1 3
Enter P3(x3,y3):
```

세 번째 좌표 입력을 요구한다.

```
C:\Windows\system32\cmd.exe
Enter P1(x1,y1): -2 5
Enter P2(x2,y2): 1 3
Enter P3(x3,y3): -1 0
```

공백을 기준으로 좌표값을 입력한다.



```
C:\Windows\system32\cmd.exe
Enter P1(x1,y1): -2 5
Enter P2(x2,y2): 1 3
Enter P3(x3,y3): -1 0
The perimeter of triangle is: 12.31
The area of triangle is: 6.50
계속하려면 아무 키나 누르십시오 . . .
```

삼각형 둘레의 길이와 넓이를 출력한다.

5. 토론

- calc_dist 함수는 두 점 사이의 거리를 계산하는 것으로, 삼각형 둘레의 길이를 한 번에 구할 수 있었다.
- 이는 세 변을 각각 구한 후 더해줌으로써 삼각형 둘레의 길이를 구할 수 있었다.

6. 결론

- 본 과제는 printf와 scanf, 사용자 정의 함수, 수학 기호, 연산을 정확히 이해하고 있어야 하는 문제였다. 문제를 해결하는 과정에서 사용자 정의 함수, 수학 기호, 연산 등을 확실히 학습할 수 있게 되었다.

7. 개선방향

- 앞으로 학습하게 될 사용자 정의 함수, 포인터, 반복문/조건문 등을 통해 프로그램이 더욱 원활히 작동될 수 있도록 할 것이다.