#include <math.h>

#include <stdio.h>

/\* 点の構造体を定義します. \*/

struct point {

double x, y;

};

/\* 三角形の構造体を定義します. \*/

struct triangle {

struct point \*p1, \*p2, \*p3;

};

/\* 点の座標をファイルから読み込みます. \*/

void read\_points(char \*, struct point \*, int);

/\* 三角形を作ります. \*/

void make\_triangle(struct triangle \*, struct point \*,

struct point \*, struct point \*);

/\* 三角形の重心を計算します. \*/

struct point compute\_center(struct triangle \*);

/\* 2点間の距離を計算します. \*/

double compute\_distance(struct point \*, struct point \*);

/\* ヘロンの公式を用いて,三角形の面積を計算します. \*/

double compute\_area(struct triangle \*);

/\*

\* main関数です.

\*/

int main(void){

/\* 点の配列 \*/

struct point points[6];

/\* 三角形の配列 \*/

struct triangle triangles[2];

/\* 三角形の重心 \*/

struct point c;

/\* 三角形の面積 \*/

double area;

/\* 点の座標をファイルから読み込みます \*/

read\_points("points.txt", points, 6);

/\* 2つの三角形を作ります \*/

make\_triangle(&triangles[0],

&points[0], &points[1], &points[2]);

make\_triangle(&triangles[1],

&points[3], &points[4], &points[5]);

/\* 三角形の重心を計算・表示します \*/

c = compute\_center(&triangles[0]);

printf("Center␣of␣Gravity␣=␣(%lf,␣%lf)\n", c.x, c.y);

c = compute\_center(&triangles[1]);

printf("Center␣of␣Gravity␣=␣(%lf,␣%lf)\n", c.x, c.y);

/\* 三角形の面積を計算・表示します \*/

area = compute\_area(&triangles[0]);

printf("Area␣=␣%lf\n", area);

area = compute\_area(&triangles[1]);

printf("Area␣=␣%lf\n", area);

return 0;

}

/\*

\* 点の座標をファイルから読み込みます.

\*/

void read\_points(char \*file\_name, struct point \*points, int n){

int i;

FILE \*fp;

fp = fopen(file\_name, "r");

for (i = 0; i < n; i++) {

fscanf(fp, "%lf␣%lf", &points->x, &points->y);

/\* 次の点を参照します \*/

points++;

}

fclose(fp);

}

/\*

\* 三角形を作ります.

\*/

void make\_triangle(struct triangle \*t, struct point \*p1,struct point \*p2, struct point \*p3){

t->p1 = p1;

t->p2 = p2;

t->p3 = p3;

}

/\*

\* 三角形の重心を計算します.

\*/

struct point compute\_center(struct triangle \*t){

struct point c;

c.x = (t->p1->x + t->p2->x + t->p3->x) / 3;

c.y = (t->p1->y + t->p2->y + t->p3->y) / 3;

return c;

}

/\*

\* 2点間の距離を計算します.

\*/

double compute\_distance(struct point \*p1, struct point \*p2){

double x = p2->x - p1->x;

double y = p2->y - p1->y;

return sqrt(x \* x + y \* y);

}

/\*

\* ヘロンの公式を用いて,三角形の面積を計算します.

\*/

double compute\_area(struct triangle \*t)

{

double a = compute\_distance(t->p1, t->p2);

double b = compute\_distance(t->p2, t->p3);

double c = compute\_distance(t->p3, t->p1);

double s = (a + b + c) / 2;

return sqrt(s \* (s - a) \* (s - b) \* (s - c));

}

実行結果

Center␣of␣Gravity␣=␣(0.000000,␣0.000000)

Center␣of␣Gravity␣=␣(3.333333,␣0.000000)

Area␣=␣0.000000

Area␣=␣0.000000