**ponto.h:**

typedef struct ponto Ponto;

Ponto\* pto\_cria(float x, float y);

void pto\_libera(Ponto \*p);

float pto\_dis(Ponto \*p1, Ponto \*p2);

**ponto.c**

#include <stdlib.h>

#include <math.h>

#include "ponto.h"

struct ponto{

    float x,y;

};

Ponto\* pto\_cria(float x, float y){

    Ponto \*p = (Ponto \*) malloc(sizeof(Ponto));

    if(p != NULL){

        p->x = x;

        p->y = y;

    }

    return p;

}

void pto\_libera(Ponto \*p){

    free(p);

}

float pto\_dis(Ponto \*p1, Ponto \*p2){

    float dx = p1->x - p2->x;

    float dy = p1->y - p2->y;

    return sqrt(dx\*dx + dy\*dy);

}

**main.c**

#include <stdio.h>

#include <stdlib.h>

#include "ponto.h"

int main(){

    float distancia;

    float x1, x2, y1, y2;

    Ponto \*p1, \*p2;

    printf("Digite as coordenadas x e y do ponto P1: ");

    scanf("%f %f", &x1, &y1);

    printf("Digite as coordenadas x e y do ponto P2: ");

    scanf("%f %f", &x2, &y2);

    p1 = pto\_cria(x1,y1);

    p2 = pto\_cria(x2,y2);

    distancia = pto\_dis(p1,p2);

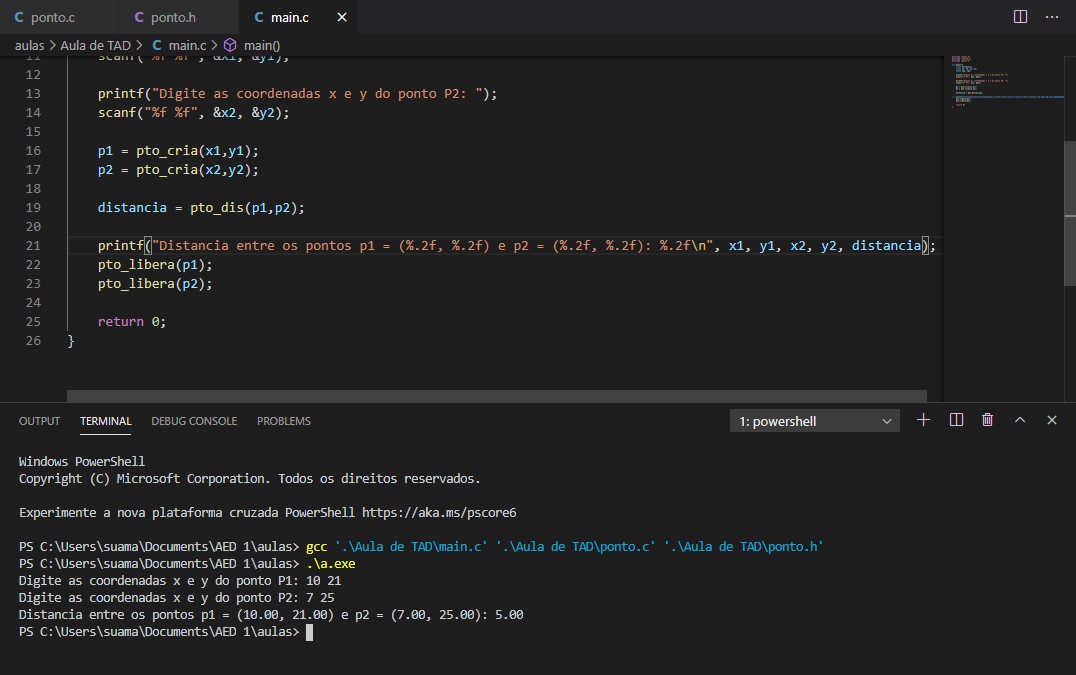
    printf("Distancia entre os pontos p1 = (%.2f, %.2f) e p2 = (%.2f, %.2f): %.2f\n", x1, y1, x2, y2, distancia);

    pto\_libera(p1);

    pto\_libera(p2);

    return 0;

}

****