

SOLUCIÓN DEL EJEMPLO DE PROBLEMA DE PROGRAMACIÓN –
SEPTIEMBRE 2010

Escribir un tipo abstracto de dato con fichero de interfaz y fichero de implementación que represente un punto en el plano con las siguientes operaciones:

1. Girar (Punto, Cantidad de grados a rotar)
2. Desplazar (Punto, Desplazamiento_eje_X,Desplazamiento_eje_Y).

Ejemplo: Sea P1(1, 0), Girar(P1,90) devuelve el P1 como (0,1), y Desplazar(P1,3,4) devuelve P1 como (4,4)

<i>Interfaz:</i>	<pre>#pragma once typedef struct TipoPunto { float x,y; void Girar(float g); void Desplazar(float dx,float dy); };</pre>
------------------	---

<i>Implementación:</i>	<pre>#include <math.h> #include <stdio.h> #include "TipoPunto.h" const float Pi = 3.14159265; typedef struct TipoPuntoPolares { float distancia,angulo; }; TipoPuntoPolares PasoPolar(float px, float py) { TipoPuntoPolares p_aux; p_aux.angulo = (180.0/Pi) * atanf (py / px); p_aux.distancia = sqrtf (px * px + py * py); return p_aux; }</pre>
------------------------	---

```
TipoPunto PasoOrtogonal(TipoPuntoPolares p_aux) {
    TipoPunto p;
    p.x = p_aux.distancia * cosf( p_aux.angulo);
    p.y = p_aux.distancia * sinf( p_aux.angulo);
    return p;
}
```

```
void TipoPunto::Girar(float g) {
    TipoPuntoPolares p_aux;
    TipoPunto p;

    p_aux = PasoPolar(x,y);
    p_aux.angulo = p_aux.angulo + g;
    p = PasoOrtogonal(p_aux);

    x = p.x;
    y = p.y;
}
```

```
void TipoPunto::Desplazar(float dx,float dy) {
    x = x + dx;
    y = y + dy;
}
```