

# Computación Gráfica

Melvin Salcedo Almiron

September 2018

## 1 ejercicio 1.6

- -1 a +1 (coordenadas normalizadas centradas)

---

```
1      inp_to_ndc( float dcx, float dcy ) {
2          ndcx = dcx / ndhml;
3          ndcy = dcy / ndvml;
4      }
5
6      ndc_to_user( float ndcx, float ndcy ) {
7          x = ( ndcx * (1 - (-1)) ) / - (-1);
8          y = ( ndcy * (1 - (-1)) ) / - (-1);
9      }
10
11     user_to_ndc( float x, float y ) {
12         ndcx = (x - (-1)) / (1 - (-1));
13         ndcy = (y - (-1)) / (1 - (-1));
14     }
15
16     ndc_to_dc( float ndcx, float ndcy ) {
17         dcx = round(ndcx * ndhml);
18         dcy = round(ndcy * ndvml);
19     }
```

---

- 0 a 100

---

```
1      inp_to_ndc( float dcx, float dcy ) {
2          ndcx = dcx / ndhm100;
3          ndcy = dcy / ndvml;
4      }
5
6      ndc_to_user( float ndcx, float ndcy ) {
7          x = ( ndcx * (100 - (0)) ) / - (0);
8          y = ( ndcy * (100 - (0)) ) / - (0);
9      }
```

```

10
11     user_to_ndc( float x, float y ) {
12         ndcx = (x - (0)) / (100 - (0));
13         ndcy = (y - (0)) / (100 - (0));
14     }
15
16     ndc_to_dc( float ndcx, float ndcy ) {
17         dcx = round(ndcx * ndhml);
18         dcy = round(ndcy * ndvml);
19     }

```

---

## 2 ejercicio 2.4

**2.0.1** Escreva um programa (em C) que desenhe um poligono regular (ou seja, um poligono com lados de comprimentos iguais e angulos internos iguais) com raio igual a um terço da altura da tela, e centro no centro da tela. O programa deve solicitar o numero de vertices, n.

- solution : <https://github.com/MelvinSalcedo/ComputacionGrafica/tree/master/grafica/ejercicios%20del%20libro%201.6%2C%202.4>

**2.1** Modifique o programa do exercicio anterior para solicitar tambien o raio r do circulo que circunscreve o poligono.

- solution : <https://github.com/MelvinSalcedo/ComputacionGrafica/tree/master/grafica/ejercicios%20del%20libro%201.6%2C%202.4>

**2.2** Alguns dispositivos vetoriais oferecem um conjunto de tres primitivas graficas:

- pen\_up
- pen\_down
- locate(dcx,dcy)

---

```

1     pen_up() {
2         moveto(dcx, dcy)
3         drawto(dcx, dcy)
4         moveto(dcx, dcy)
5         drawto(dcx, dcy)
6     }
7
8     pen_down() {

```

```

9          drawto(dcx,dcy)
10         drawto(dcx,dcy)
11     }
12
13     locate(dcx, dcy) {
14         moveto(dcx, dcy);
15     }

```

---

### 2.3 Calcule as razoes de aspecto (grafica e fisica), e as resolucoes de area horizontal e vertical de uma tela de TV colorida padrao, onde

- ancho = 42 cm
- alto = 31 cm
- ndh = 546
- ndv = 434

1. resolucion horizontal =  $546 / 420$ ;
2. resolucion vertical =  $434 / 310$ ;
3. total de puntos direccionables =  $546 * 434$ ;
4. resolucion de area = total de puntos direccionables /  $(420/546)$ ;
5. razon de aspecto =  $(310/546) / (420/434)$ ;
6. razon de aspecto fisico =  $310 / 420$ ;