

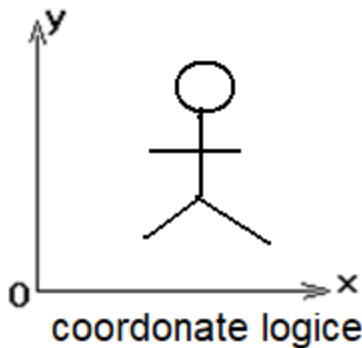
Transformarea de vizualizare 2D

Prof. univ. dr. ing. Florica Moldoveanu

Curs Elemente de Grafică pe Calculator – UPB, Automatică și Calculatoare
2020-2021

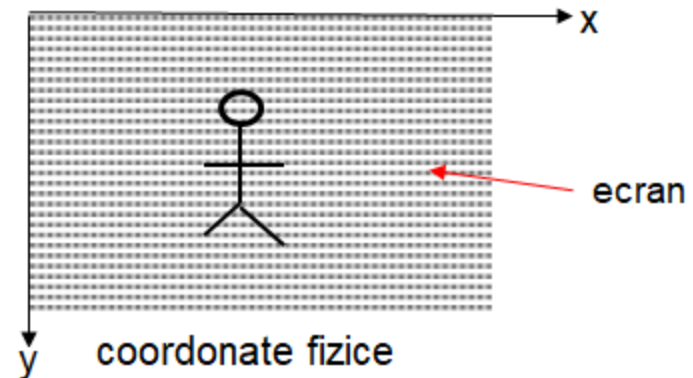
Coordonate logice și coordonate fizice

- ❑ Desenele reprezentate într-un program de aplicatie grafica 2D sunt, de regula, raportate la un sistem de coordonate diferit de cel al suprafetei de afisare.
- ❑ Exemple: planul unei case, un desen tehnic, graficul unei functii, etc.
- Editorul grafic trebuie sa permita definirea desenului intr-un sistem de coordonate logice.



(metri, milimetri, viteza, timp, etc.)

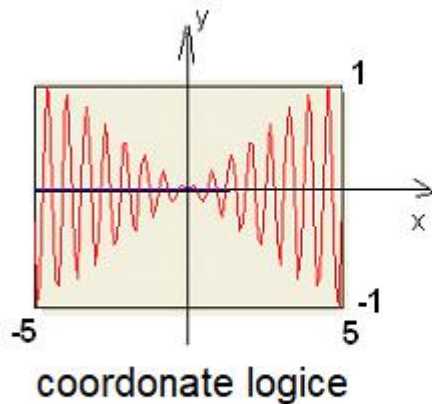
Sistemul de coordonate utilizator



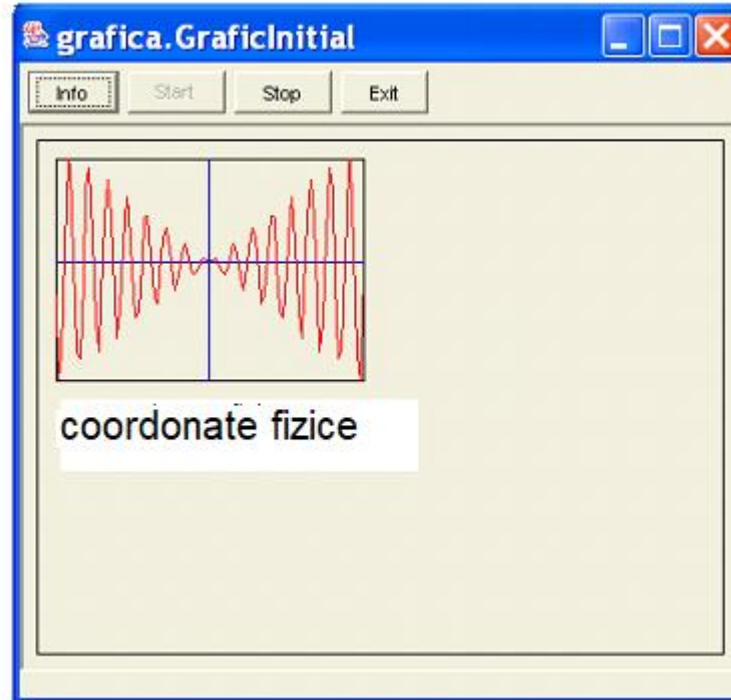
(pixeli, milimetri, s.a.)

Sistemul de coordonate dispozitiv

Transformarea coordonate logice - coordonate fizice



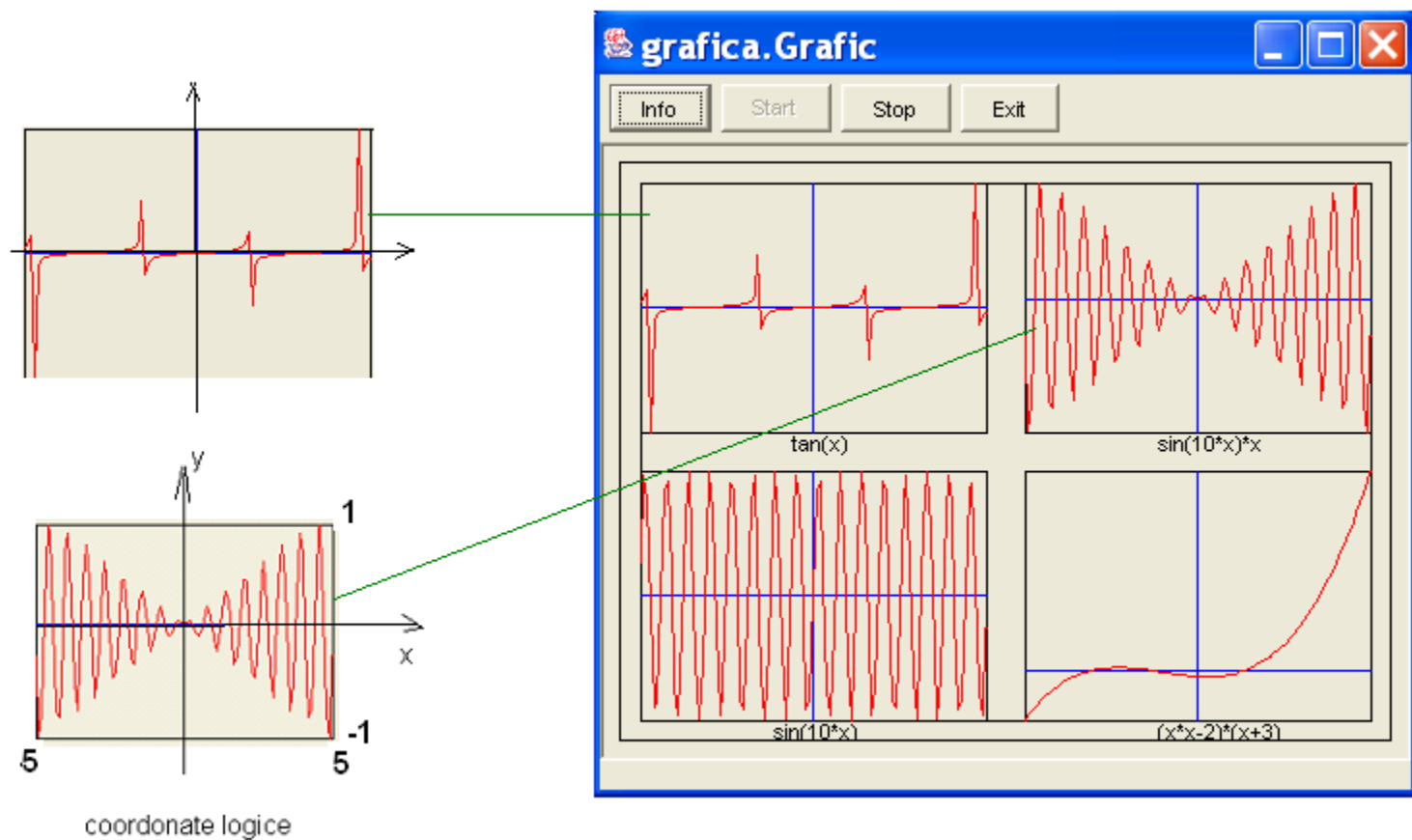
Aplicatia defineste
graficul functiei in
sistemul de coordonate
utilizator (logice).



Coordonatele punctelor de pe graficul afisat
pe ecran sunt adrese de pixeli.

Coordonatele logice trebuie transformate in coordonate fizice.

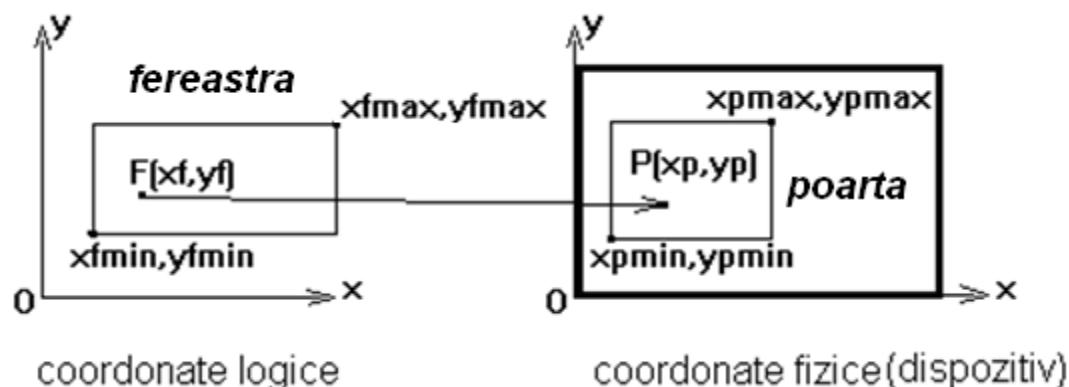
Transformarea de vizualizare 2D (1)



Este necesara o transformare generala, prin care un desen definit in coordonate logice sa poata fi afisat intr-o zona oarecare a ecranului.

Transformarea de vizualizare 2D (2)

- transformarea “fereastră-poarta”-



- fereastră: zona din spatiul coord. logice al carei continut se va afisa
- poarta: zona din spatiul coord. fizice in care se va afisa continutul ferestrei

▪ Transformarea este definita folosind 2 dreptunghiuri, definite in cele 2 sisteme de coordonate, numite: **fereastră (de vizualizare) –vizorul** și **poarta (de afisare)**.

• Transformarea se mai numeste **transformarea fereastră-poarta (window-viewport)**

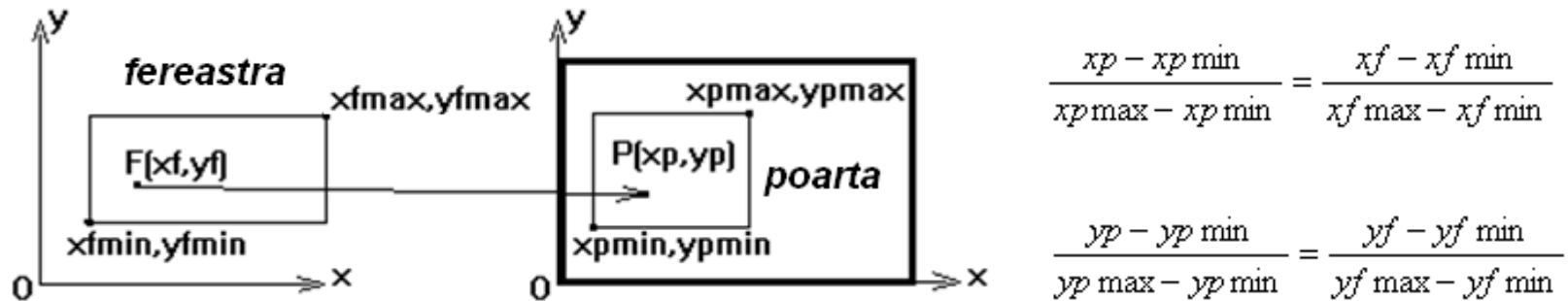
F: un punct din fereastră.

P: punctul in care se transforma F prin transformarea fereastră-poarta.

➤ Transformarea fereastră-poarta trebuie sa asigure ca:

“ pozitia relativa a lui P in poarta de afisare este aceeași cu pozitia relativa a lui F in fereastră”.

Transformarea de vizualizare 2D (3)



$$sx = \frac{xp_{\max} - xp_{\min}}{xf_{\max} - xf_{\min}}$$

$$sy = \frac{yp_{\max} - yp_{\min}}{yf_{\max} - yf_{\min}}$$

- sx, sy reprezintă factorii de scalare ai transformării
- tx, ty depind de pozițiile celor 2 dreptunghiuri față de originea sistemului de coordonate în care sunt definite

și $tx = xp_{\min} - sx * xf_{\min}$ $ty = yp_{\min} - sy * yf_{\min}$

$$\begin{aligned} xp &= xf * sx + tx \\ yp &= yf * sy + ty \end{aligned}$$

← Exprimarea matematică a transformării fereastră – poartă, considerând o aceeași orientare a axelor celor 2 sisteme de coordonate.

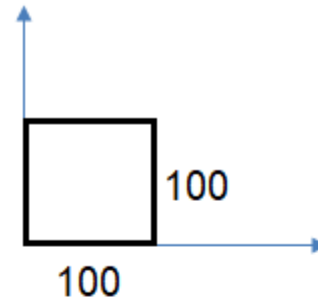
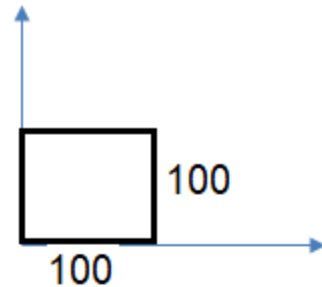
Transformarea de vizualizare 2D (4)

$$sx = \frac{xp_{max} - xp_{min}}{xf_{max} - xf_{min}}$$

$$sy = \frac{yp_{max} - yp_{min}}{yf_{max} - yf_{min}}$$

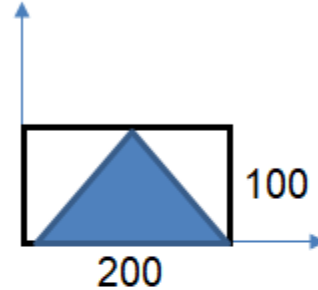
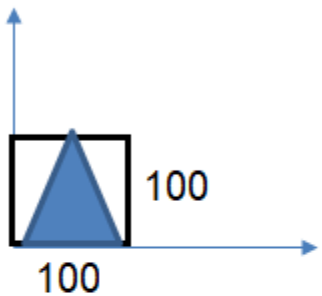
$$tx = xp_{min} - sx * xf_{min}$$

$$ty = yp_{min} - sy * yf_{min}$$



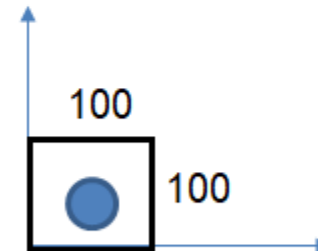
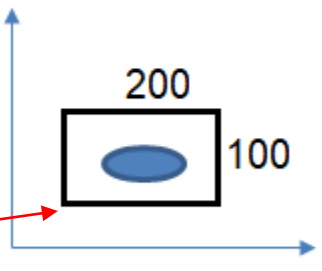
$$sx=1, sy = 1$$

$$tx = 0, ty = 0$$



$$sx=2, sy = 1$$

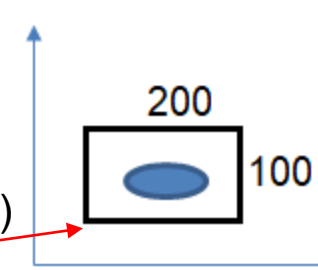
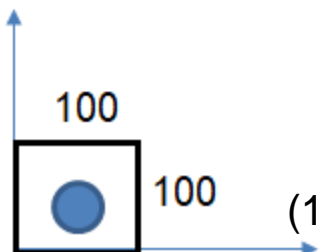
$$tx = 0, ty = 0$$



$$sx=0.5, sy = 1$$

$$tx = 0 - 0.5 * 10 = -5$$

$$ty = 0 - 1 * 20 = -20$$



$$sx=2, sy = 1$$

$$tx = 10$$

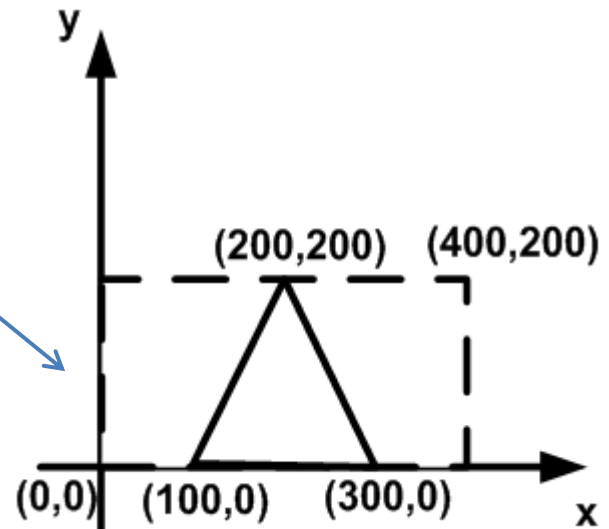
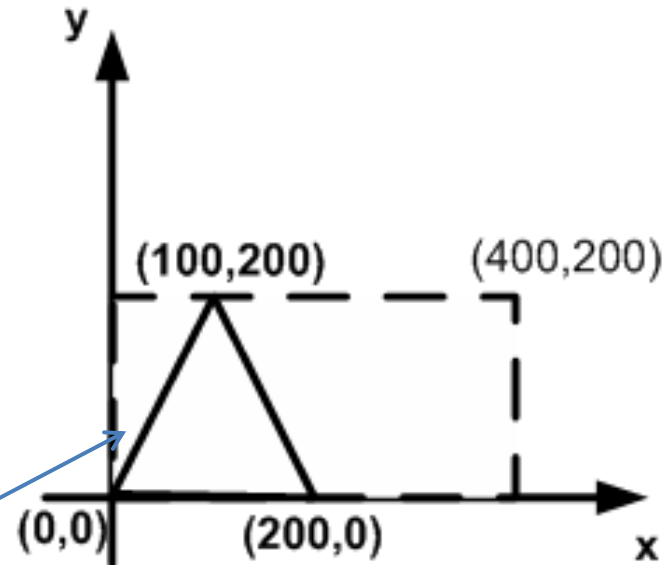
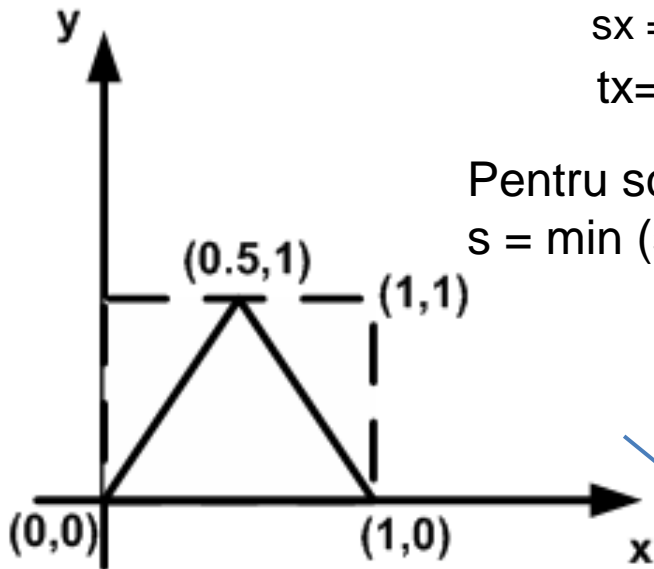
$$ty = 20$$

Transformarea de vizualizare 2D (5)

**Scalare uniforma
cu afisare centrata in poarta de vizualizare**

$sx = 400, sy = 200$
 $tx=0, ty=0$

Pentru scalare uniforma:
 $s = \min (sx, sy)$



$tsx = 100,$
 $t sy = 0$

$$tsx = (xpmax - xpmin - s*(xfmax - xfmin))/2$$

$$= (400 - 200*1) / 2 = 100$$

$$tsy = (ypmax - ypmin - s*(yfmax - yfmin))/2$$

$$= (200 - 200*1) / 2 = 0$$

Transformarea de vizualizare 2D (7)

Efectele transformarii

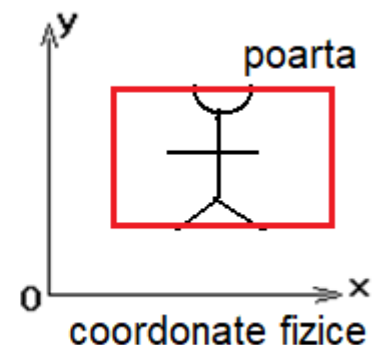
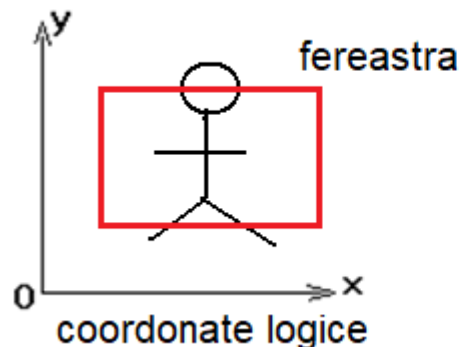
- Marire / micșorare, în funcție de dimensiunile ferestrei și ale porții
- Deformare: fereastra și poarta nu sunt dreptunghiuri asemenea
- Pentru scalare uniformă: $s = \min(s_x, s_y)$
- Afișare centrată în poartă: translație suplimentară pe axa ox sau pe axa oy :

$$ts_x = (x_{pmax} - x_{pmin} - s * (x_{fmax} - x_{fmin})) / 2$$

$$ts_y = (y_{pmax} - y_{pmin} - s * (y_{fmax} - y_{fmin})) / 2$$

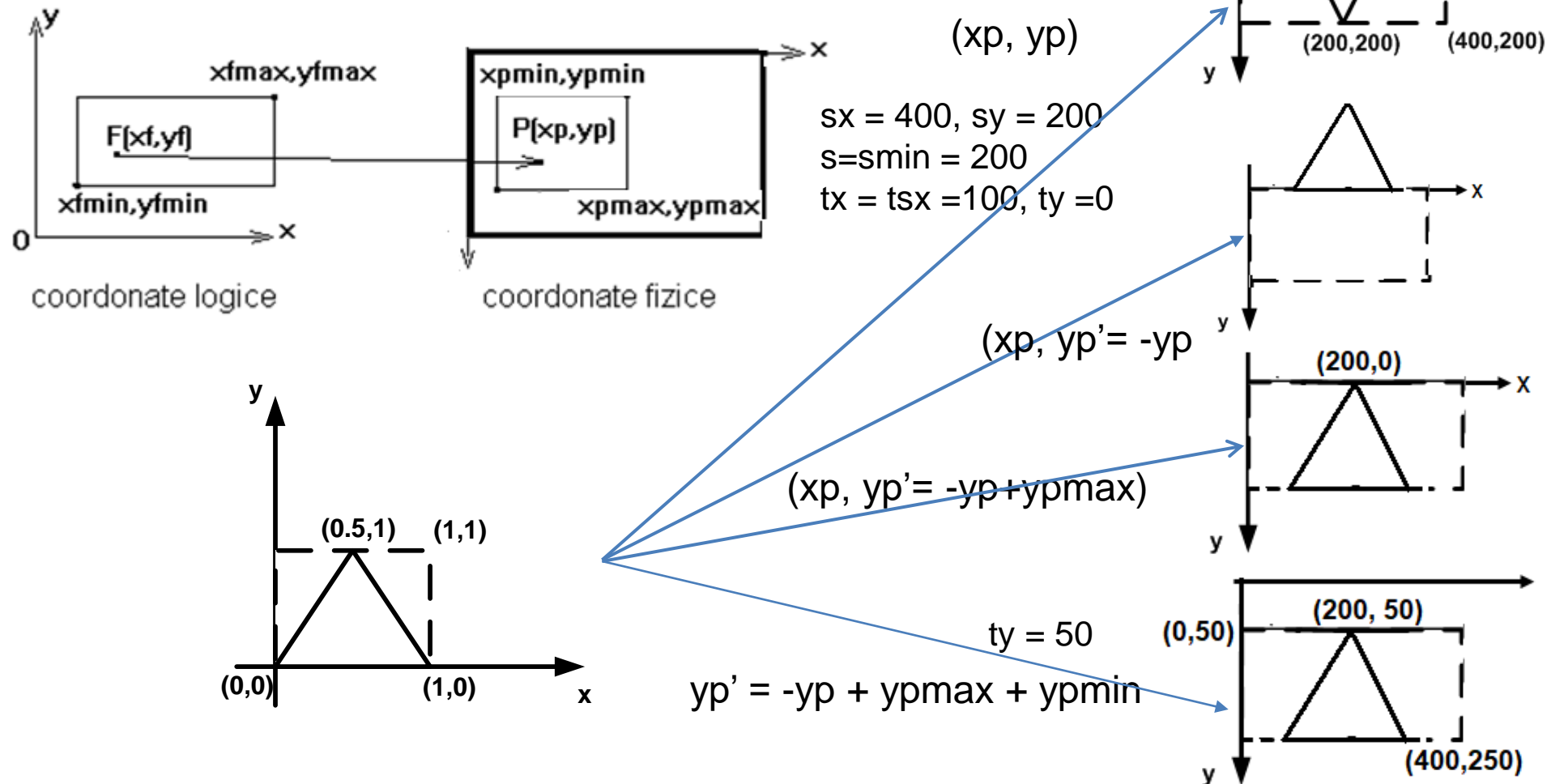
Translația totală: $tx + ts_x, ty + ts_y$

- Decuparea primitivelor aflate în afara ferestrei vizuale



Transformarea de vizualizare 2D (8)

Corectia coordonatei yp atunci cand axa OY a sist. coord. fizice este orientata in jos



Transformarea de vizualizare 2D (9)

Formulele transformarii fereastra – poarta atunci cand axa OY a sistemului de coordonate fizice este orientata in jos:

$$xp = xf * sx + tx$$

$$yp = -(yf * sy + ty) + ypmin + ypmax = -yf * sy + ypmin + ypmax - ty$$

Exprimarea matriciala a transformarii fereastra-poarta in cazul corectiei coordonatei y

$$\begin{bmatrix} xp \\ yp \\ 1 \end{bmatrix} = M * \begin{bmatrix} xf \\ yf \\ 1 \end{bmatrix}$$

$$M = \begin{bmatrix} sx & 0 & tx \\ 0 & -sy & ypmin + ypmax - ty \\ 0 & 0 & 1 \end{bmatrix}$$

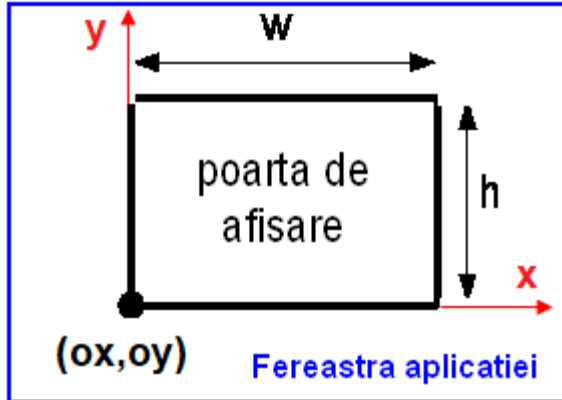
Implementarea transformarii de vizualizare în OpenGL (1)

- Pentru setarea zonei de afisare, în OpenGL se apeleaza

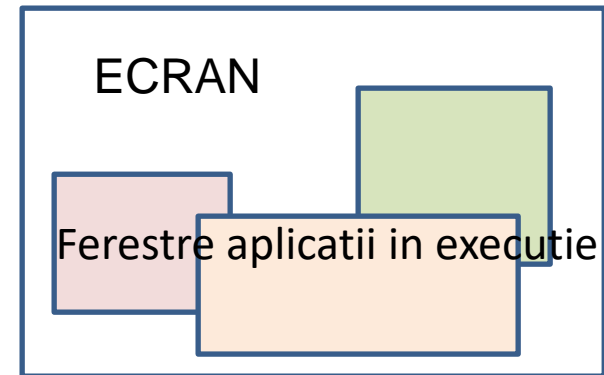
`glViewport(ox, oy, width, height);`

unde (ox,oy) sunt coord. coltului stanga jos al viewportului, in coordonatele ferestrei aplicatiei.

- Orientarea axelor viewportului este aceeași cu a sistemului coordonatelor logice:
 - Nu este necesara corectia coordonatei y → matricea transformarii de vizualizare este:



$$M = \begin{bmatrix} sx & 0 & tx \\ 0 & sy & ty \\ 0 & 0 & 1 \end{bmatrix}$$



- Se poate defini un desen in coordonate logice.
- Pentru obtinerea desenului in coordonate ecran se foloseste ca matrice de transformare a vârfurilor - matricea transformării de vizualizare