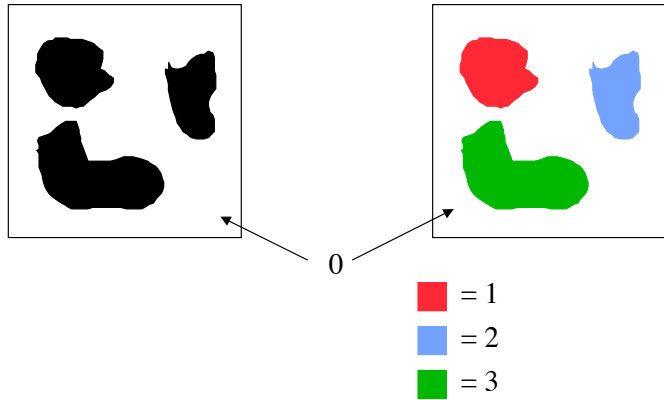
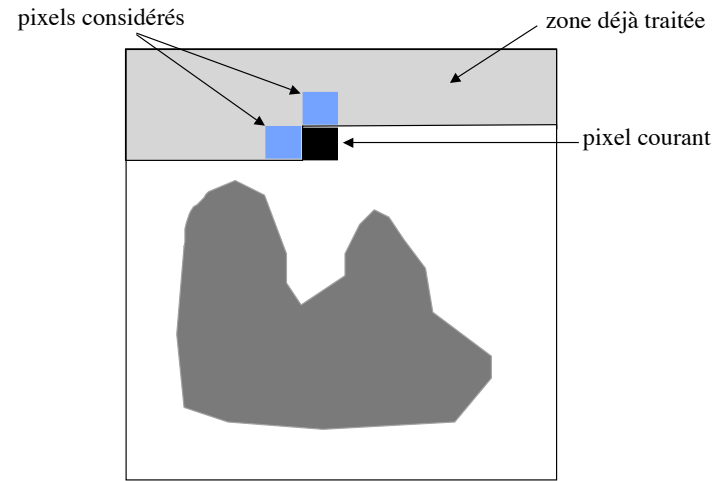


# ETIQUETAGE

Objectif :

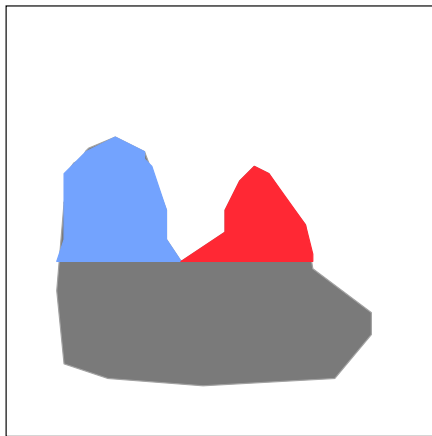


# ETIQUETAGE : traitement local et séquentiel

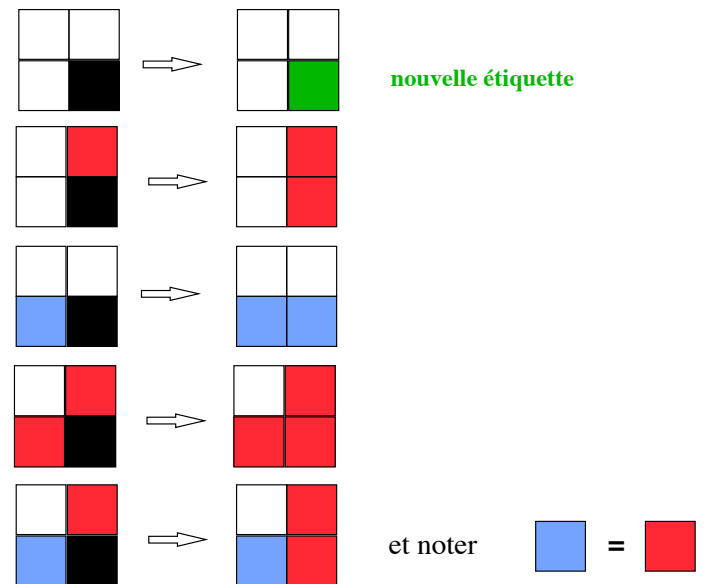


# ETIQUETAGE

problème des étiquettes synonymes



# ETIQUETAGE : Principe de propagation



## ETIQUETAGE : algorithme de propagation

- Coloriage des tâches** (méthode de Ballard et Brown)  
l'algorithme examine les pixels au-dessus  $X_h$  et à gauche  $X_g$  du pixel courant  $X_c$  :  

$$\begin{matrix} & X_h \\ X_g & X_c \end{matrix}$$
 Soit  $f$  la fonction image.

```

débüt
si f(Xc) = 0 alors continue
sinon débüt
    si f(Xh) = 1 et f(Xg) = 0 alors coul(Xc) ← coul(Xh)
    si f(Xg) = 1 et f(Xh) = 0 alors coul(Xc) ← coul(Xg)
    si f(Xg) = 1 et f(Xh) = 1 alors débüt
        coul(Xc) ← coul(Xh)
        équivalent (coul(Xg) , coul(Xh))
    fin
    si f(Xg) = 0 et f(Xh) = 0 alors débüt
        coul(Xc) ← k
        k ← k+1
    fin

```

Etendre cet algorithme aux 8 voisins :  

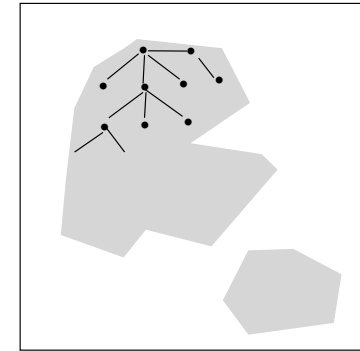
$$\begin{matrix} X1 & X2 & X3 \\ X4 & Xc \end{matrix}$$

- Renommer les couleurs équivalentes

## ETIQUETAGE : algorithme récursif

### Principe :

- rechercher le début d'une forme et lui attribuer un nouveau numéro
- propager à l'intérieur de la forme
- passer à la forme suivante
- ...



- Éviter les bouclages
- Gérer la récursivité par une file

## ETIQUETAGE : algorithme récursif

```

Niveau 1 Etiquetage(T) // T tableau image de pixels valant "fond" ou "forme"
{ no=init; // initialisation des numéros d'étiquettes
  for (i=0; i< NL; i++) // parcours de l'image
    for (j=0; j< NC; j++)
      if (forme(T,i,j)) // début d'une forme
        { RemplirObj(T,i,j,no) // remplissage de la forme
          no++; } // rq : valeurs de no ≠ "fond" ou "forme"
}

```

```

Niveau 2 RemplirObj(T,i,j,n) //propage l'étiquette n à la forme débutant en T(i,j)
{ initialiserFile(F); // F file vide
  addFile(F,i,j); // (i,j) en queue de file, pour propagation
  T[i][j] = n; // (i,j) marqué donc ≠ "forme"
  tantque( fileNonVide(F) ) {
    extraireFile(F,x,y); // (x,y) ← tête de file
    if (testBord(x-1,y-1) && forme(T,x-1,y-1))
      { addFile(F,x-1,y-1); T[x-1][y-1]= n }
    // ..... idem pour chaque pixel voisin de (x,y)
  }
}

```

## ETIQUETAGE

Codage par plages  
 $(x_0, l_0) (x_1, l_1) \dots (x_n, l_n)$

x	x	x	x	x	x	x	x	x	x		x	x	x		x	x	x	x	x		
x	x	x	x	x	x	x	x	x	x		x	x		x	x	x	x	x			
x	x					x	x				x	x	x	x	x						

Calcul des recouvrements de plages  
 Affectation d'étiquette à tous les éléments d'une plage