Épreuve d'informatique de l'X - 2008 - PSI

Lissage de vecteurs obliques

Question 1 : On parcourt tous les pixels de l'écran par une double boucle et on peint tous les pixels de coordonnées (x, y) vérifiant $\lfloor ax + b \rfloor \leqslant y$.

```
peindreRegion := proc (w, h, a, b)
local x, y;
for x from 0 to w−1 do
  for y from 0 to h−1 do
    if partieEntière(a∗x+b) <= y then peindrePixel(x, y, 1) fi
  od;
od;
end;
```

Question 2 : Pour chaque ordonnée x, on peint les points de coordonnées (x, y) pour y compris entre $ax + b$ et $h - 1$.

```
partieEntière := floor;

peindreRegionbis := proc (w, h, a, b)
local yMin, x, y;
yMin := proc (x)
    return partieEntière(a∗x+b)
end;
for x from 0 to w−1 do
    for y from yMin(x) to h−1 do
        peindrePixel(x, y, 1);
    od;
od;
end;
```

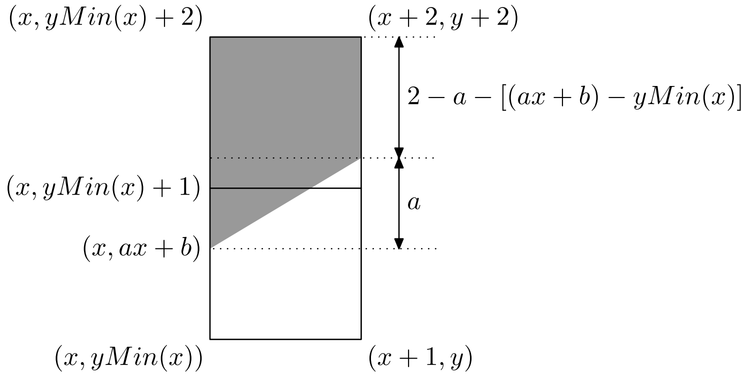
Question 3 : On parcourt tous les pixels de l'écran par une double boucle et on peint tous les pixels de coordonnées (x, y) vérifiant $\lfloor ax + b \rfloor \leqslant y$. Pour chaque pixel, on calcule l'intensité de gris à utiliser (double boucle des lignes 6 à 12).

```
1  peindreRegionAA := proc (w, h, a, b)
2  local g, x, y, i, j;
3  for x from 0 to w−1 do
4     for y from 0 to h−1 do
5        g := 0;
6        for i from 0 to 1 do
7            for j from 0 to 1 do
8               if a∗(x+.5)+b < y+.5∗j
9                   then g := g+.25
10              fi;
11           od;
12       od;
13       peindrePixel(x, y, g)
14    od;
15 od;
16 end;
```

Question 4 : On ne regarde que les pixels de coordonnées (x, y) vérifiant $ax + b \leqslant y$ et on ne traite de façon particulière que ceux qui vérifient $ax + b - 0.5 \leqslant y \leqslant a(x + 0.5) + b$ pour calculer l'intensité du gris.

```
peindreRegionAAbis := proc (w, h, a, b)
local yMin, g, x, y, f, i, j;
yMin := proc (x)    return partieEntière(a∗x+b) end;
for x from 0 to w−1 do
   for y from yMin(x) to h−1 do
      g := 0.;
      if a∗(x+.5)+b <= y
         then g := 1
         else for i from 0 to 1 do
                 for j from 0 to 1 do
                     if a∗(x+.5∗i)+b < y+.5∗j
                        then g := g+.25
                     fi;
                 od;
              od;
      fi;
      peindrePixel(x, y, g)
   od;
od;
end;
```

Question 5 :



En reprenant les données du dessin ci-dessus, $S(x)$ apparaît comme la somme des aires d'un triangle et d'un rectangle dont l'un des cotés a pour mesure 1 et l'autre respectivement a et $2-a-[(ax+b)-yMin(x)]$ ce qui nous donne :

$$S(x) = \frac{a}{2} + 2 - a - [(ax+b) - yMin(x)] = 2 - \frac{a}{2} - (ax+b) + yMin(x)$$

$$\Rightarrow \boxed{S(x) = 2 - \frac{a}{2} - ax - b + yMin(x)}$$

Question 6 : Lignes 3 à 7, on écrit la fonction f donnée par l'énoncé. Ligne 8, on calcule ce qui correspond à $yMin(0)$ et ligne 9 ce qui correspond à $S(0)$. La boucle des lignes 10 à 21 permet de traiter tous les points d'abscisses x. Pour les points d'abscisses x, on ne peint que les points d'ordonnées comprises entre $yMin(x)$ et $h-1$, $yMin(x)$ étant représenté par la variable `yMin`. On commence par peindre les pixels noirs (boucle des lignes 11 à 13). Puis on traite le cas des points à la frontière qui vont être peints en gris (lignes 14 à 16) en prenant soin de vérifier qu'ils sont dans la fenêtre graphique. On calcule $S(x+1)$ et $yMin(x+1)$ représentés par les variables respectives `S` et `yMin` (lignes 17 à 20) en suivant les indications données par les tableaux de la page 4 de l'énoncé.

```
1  peindreRegionAAA := proc (w, h, a, b)
2  local f, s0, S, x, y, yMin;
3  f := proc (x)
4  local temp;
5    temp := x-1.+a/2.;
6    return temp*temp*(1/2)/a
7  end proc;
8  yMin := partieEntière(b);
9  S := 2. - a/2 - b + partieEntière(b);
10 for x from 0 to (w-1) do
11      for y from yMin+2 to h-1 do
12              peindrePixel(x, y, 1.)
13      od;
14      if S <= 1.+a/2. then s0 := f(S) else s0 := S-1. fi;
15      if  yMin  < h then peindrePixel(x, yMin, s0) fi;
16      if yMin+1 < h then peindrePixel(x, yMin+1, S-s0) fi;
17      if S <= 1.+a/2.
18                then S := 1.+S-a; yMin := yMin+1
19                else S := S-a
20      fi;
21 od;
22 end;
```

Question 7 : En posant $\alpha = S - 1 + \dfrac{a}{2}$, on a :

$f(S) = f0 = \dfrac{1}{2a}\alpha^2$, $\qquad f'(S) = \dfrac{1}{a}\left(S - 1 + \dfrac{a}{2}\right) = \dfrac{1}{a}\alpha = f1$,

$f2 = -a.f'(S) = -\alpha$. D'où :

$$f(S+1) - f(S) = \frac{1}{2a}((\alpha+1)^2 - \alpha^2) = \frac{1}{2a}(2\alpha+1) = \frac{1}{a}\alpha + \frac{1}{2a} = f1 + \frac{1}{2a}$$

$$\Rightarrow \boxed{f(S+1) = f0 + f1 + \frac{1}{2a}}$$

Comme f' est un polynôme de degré 1 de coefficient dominant $\dfrac{1}{a}$, on a :

$$f'(S+1) = f'(S) + \frac{1}{a} \Rightarrow \boxed{f'(S+1) = f1 + \frac{1}{a}}$$

$$\text{et } -af'(S+1) = -af'(S) - 1 \Rightarrow \boxed{-af'(S+1) = f2 - 1}$$

$$f(S-a) - f(S) = \frac{1}{2a}((\alpha-a)^2 - \alpha^2) = \frac{1}{2a}(2\alpha - a).(-a) = -\alpha + \frac{a}{2} = f2 + \frac{a}{2}$$

$$\Rightarrow \boxed{f(S-a) = f0 + f2 + \frac{a}{2}}$$

Comme f' est un polynôme de degré 1 de coefficient dominant $\dfrac{1}{a}$, on a :

$$f'(S-a) = f'(S) - 1 \Rightarrow \boxed{f'(S-a) = f1 - 1}$$

$$\text{et } -af'(S-a) = -af'(S) - a \Rightarrow \boxed{-af'(S-a) = f2 + a}$$

Question 8 : On commence par calculer une fois pour toutes les quantités $\dfrac{1}{a}$, $\dfrac{1}{2a}$, $\dfrac{a}{2}$ (lignes 3 à 5). La variable `minimum` correspond à `yMin(x)` pour le x courant. On initialise les variables `f0`, `f1` et `f2` (ligne 9-11). Dans la boucle des lignes 12-36, on traite les différentes abscisses possibles. Lignes 13-15, on traite les pixels noirs. Lignes 16-35, on traite les pixels du bord de la région $\mathcal{T}$. On traite les deux cas $S(x) \leqslant 1 + \frac{a}{2}$ (lignes 17-23) et $S(x) > 1 + \frac{a}{2}$ (lignes 24-27) Dans le premier cas, on calcule `f0`, `f1` et `f2` qui correspondent à $f(1 + S(x) - a) = f(S(x+1))$ en deux étapes, en utilisant les propriétés démontrées à la question 8. Dans le deuxième cas, on calcule `f0`, `f1` et `f2` qui correspondent à $f(S(x) - a) = f(S(x+1))$ en une étape. Lignes 29-30, on peint les deux pixels à la limite du demi plan avec une couleur proportionnelle à la surface couverte pas le demi plan.

```
1  peindreRegionAAAbis :=proc (w, h, a, b)
2  local s0, inv_a, inv_a_div_2, a_sur_2,
3        yMin, S, f0, f1, f2, x, y;
4  inv_a := 1/a;
5  inv_a_div_2 := 1./((2.*a));
6  a_sur_2 := a/2.;
7  yMin := partieEntière(b);
8  S := 2. - a/2. - b + yMin;
9  alpha = S - 1. + a_sur_2;
10 f0 := alpha*alpha/(2.*a);
11 f1 := alpha/a;
12 f2 := -alpha;
13 for x from 0 to w-1 do
14    for y from yMin+2 to h-1 do
15       peindrePixel(x, y, 1.)
16    od;
17    if S <= 1.+a_sur_2
18       then s0 := f0;
19            f0 := f0+f1+inv_a_div_2;
20            f1 := f1+inv_a;
21            f2 := f2 -1.;
22            f0 := f0+f2+a_sur_2;
23            f1 := f1 -1.;
24            f2 := f2+a
25       else s0 := S-1.;
26            f0 := f0+f2+a_sur_2;
27            f1 := f1 -1.;
28            f2 := f2+a
29    fi;
30    if  yMin  < h then peindrePixel(x, yMin, s0) fi;
31    if yMin+1 < h then peindrePixel(x, yMin+1, S-s0) fi;
32    if S <= 1.+a_sur_2
33       then S := 1.+S-a;
34            yMin := yMin+1
35       else S := S-a
36    fi;
37 od;
38 end;
```