```
/*inspiré d'un programme de Guillaume Connan */
//size(40mm,0);
unitsize(20cm);

void triangleb(path p, int n, int coul);
void triangleh(path p, int n, int coul);

void carre(path p, int n, int coul)
{
  if (n <= 1) {
    if (coul != 0) filldraw(p, gray); else filldraw(p, white);
  } else {
    triangleb(point(p,0)--point(p,1)--point(p,3)--cycle, n-1, coul);
    triangleh(point(p,1)--point(p,3)--point(p,2)--cycle, n-1, coul);
  }
}

triangleb = new void(path p, int n, int coul)
{
  if (n <= 1) {
    if (coul != 0) filldraw(p, gray); else filldraw(p, white);
  } else {
    pair p01 = point(p,0.5);
    pair p12 = point(p,1.5);
    pair p20 = point(p,2.5);
    triangleb(point(p,0)--p01--p20--cycle, n-1, coul);
    triangleb(p01--point(p,1)--p12--cycle, n-1, coul);
    triangleb(p20--p12--point(p,2)--cycle, n-1, 1-coul);
    triangleh(p01--p20--p12--cycle, n-1, coul);
  }
};

triangleh = new void(path p, int n, int coul)
{
  if (n <= 1) {
    if (coul != 0) filldraw(p, gray); else filldraw(p, white);
  } else {
    pair p01 = point(p,0.5);
    pair p12 = point(p,1.5);
    pair p20 = point(p,2.5);
    triangleh(p20--p01--point(p,0)--cycle, n-1, coul);
    triangleh(p12--point(p,1)--p01--cycle, n-1, coul);
    triangleh(point(p,2)--p12--p20--cycle, n-1, 1-coul);
    triangleb(p12--p20--p01--cycle, n-1, coul);
  }
};

carre(unitsquare, 6, 1);
```