On the “progression transformed into art” project

Krzysztof Pszczola

Instytut Matematyki UJK, Poland, pszczola@fr.pl

BachoTeX 2015
In the project “progression transformed into art” we have graphics’ posesinging all of the following three properties:

- they visualize certain progressions (arithmetic, geometric...),
- they are generated with a short and elegant METAPOST code (that’s a working assumption, but as it happens with assumptions, the realizations might differ),
- they are visually attractive (that’s again an assumption; see above).

The web-page of the project is http://pro-trans.tumblr.com/.
color gold;
gold := (255/255,215/255,0/255);

beginfig(1);
n:=7;
u:=16bp;
r:=u;
q:=2;
width:=u;
height:=u;
sx:=0;
sy:=0;

hlh:=1bp;
pickup pensquare scaled 2;
for i=0 upto n-1:
    fill (0,0)--(width,0)--(width,height)--(0,height)--cycle
        shifted (sx,sy) withcolor gold; % bars
    draw (sx+width-hlh,sy-hlh)--(sx-hlh,sy-hlh)--(sx-hlh,sy+width-hlh); % lines
    sx:=sx+width*q; % geometric progression
    height:=height+r; % arithmetic progression
endfor;
endfig;

progression transformed into art
beginfig(2);
 n:=8;
 u:=1pc;
 x:=0;
 y:=0;
 r:=u;
 heightp:=r;
 heights:=r;
 width:=3*u;
 linewidth:=2bp;
 corr:=1/7*width;
 pickup pencircle scaled linewidth;
 for i=0 upto n-2:
  draw (x,0)--(x,heightp-corr){up}..
  (x+corr,heightp){right}--(x+width-corr,heightp)
  {right}..(x+width,heightp-corr){down}--
  (x+width,0); {one more bar}
  x:=x+2*width;
  heightp:=heightp+r; % arithmetic progression
  heights:=heights+heightp;
  % partial sum of the sequence
 endfor;
 draw (x,0)--(x,heightp-corr){up}..
  (x+corr,heightp){right}--(x+width-corr,heightp)
  {right}..(x+width,heightp-corr){down}--
  (x+width,0);
 draw (-.5*width,0)--(x+.5*width,0) withcolor blue;
 endfig;
\begin{fig}
n=11; 
u=50cm; 
correction=.75bp; 
r=.707; % ~\sqrt{2}/2 
d:=u*r; 
for i=1 upto n-1: 
fill (d,d)--(-d,d)--(-d,-d)--(d,-d)--cycle 
withcolor ((n-i)/n); 
d:=d*r; 
endfor; 
d:=u-3bp; 
draw (0,0)--(d,d) 
withpen pencircle scaled 8bp; 
draw (d,d)--(-d,d)--(-d,-d)--(d,-d)--cycle 
withpen pensquare scaled 6bp; 
draw (d,d) 
withpen pencircle scaled 2pc 
withcolor red; 
endfig;
color yellow;
yellow := (255/255,235/255,0/255);
beginfig(5);
n:=7;
u:=1pc;
azero:=u;
r:=u;
bzero:=u;
q:=2;
x:=0;
y:=0;
width:=0;
height:=0;
pickup pensquare scaled 2.4;
width:=bzero*(1-q**(n))/(1-q);
height:=(2*azero+(n)*r)*(n+1)/2;
% paint squares in red
fill (0,0)--(u,0)--(u,u)--(0,u)--cycle
withcolor red;
fill (u,u)--(3u,u)--(3u,3u)--(u,3u)--cycle
withcolor red;
fill (3u,6u)--(7u,6u)--(7u,10u)--(3u,10u)--cycle
withcolor red;
fill (7u,28u)--(15u,28u)--(15u,36u)--(7u,36u)--cycle
withcolor red;
% paint diagonal in (different shades of) blue
fill (3u,3u)--(7u,3u)--(7u,6u)--(3u,6u)--cycle
withcolor .8[blue,white];
fill (7u,6u)--(15u,6u)--(15u,10u)--(7u,10u)--cycle
withcolor .6[blue,white];
fill (15u,10u)--(31u,10u)--(31u,15u)--(15u,15u)--cycle
withcolor .4[blue,white];
fill (31u,15u)--(63u,15u)--(63u,21u)--(31u,21u)--cycle
withcolor .2[blue,white];
fill (63u,21u)--(127u,21u)--(127u,28u)--(63u,28u)--cycle
withcolor .0[blue,white];
% add some yellow accents
fill (0,21u)--(1u,21u)--(1u,28u)--(0,28u)--cycle
withcolor yellow;
fill (63u,0)--(127u,0)--(127u,1u)--(63u,1u)--cycle
withcolor yellow;
% make grid
draw (0,height)--(0,0)--(width,0);
s:=bzero;
y:=azero;
for i=0 upto n-1:
draw (x,0)--(x,height);
bzero:=bzero*q; % geometric progression
x:=x+bzero;
draw (0,y)--(width,y);
azero:=azero+r; % arithmetic progression
y:=y+azero;
endfor;
draw (0,y)--(width,y);
azero:=azero+r;
endfig;
beginfig(0);
  n:=7;
  r:=u;
  q:=2;
  widthblack:=u;
  widthwhite:=u;
  height:=8u;
  x:=0;
  y:=0;
  sep:=-1.5height;
  % just to add white margins
  maxx:=0.5height+((1+n)*n/2+(q**(n+1)-1))*u;
  fill (-0.5height,sep-0.5height)--(maxx,sep-0.5height)--(maxx,1.5height)--(-0.5height,1.5height)--cycle withcolor white;

  % drawing bars
  for i=0 upto n-1:
    fill (0,0)--(widthblack,0)--(widthblack,height)--(0,height)--cycle shifted (x,0);
    x:=x+widthblack;
    widthblack:=widthblack*q; % geometric progression
    fill (0,0)--(widthwhite,0)--(widthwhite,height)--(0,height)--cycle shifted (x,sep);
    x:=x+widthwhite;
    widthwhite:=widthwhite+r; % arithmetic progression
  endfor;

  fill (0,0)--(widthblack,0)--(widthblack,height)--(0,height)--cycle shifted (x,0);
  x:=x+widthblack;

  pickup pencircle scaled .25u;
  draw (x,sep-.125u)--(x,height-.125u);

  % text
  defaultscale:= 1.33;
  defaultfont:="pltt12";
  label.rt("for i=0 upto 6: fill (0,0)--(widthblack,0)--(widthblack,height)--(0,height)--cycle shifted (x,0); x:=x+widthblack; widthblack:=widthblack*q; fill (0,0)--(widthwhite,0)--(widthwhite,height)--(0,height)--cycle shifted (x,sep); x:=x+widthwhite; widthwhite:=widthwhite+r; endfor; fill (0,0)--(widthblack,0)--(widthblack,height)--(0,height)--cycle shifted (x,0); x:=x+widthblack; draw (x,sep-hlh)--(x,height-hlh);",(.89u,-.27height));
endfig;
There is enough material for the exhibition in a small / medium size art gallery; if someone is interested – just let me know.
The thoughts which appear while working on the project lead me to an idea of teaching basics of programming to elementary and secondary school pupils based on the METAPOST language.

The main advantage here is the almost obvious debugging: if the generated picture is not as planned, that something is wrong.

Coauthors of such a schoolbook are welcome.
Thank you for your attention

All comments are welcome.