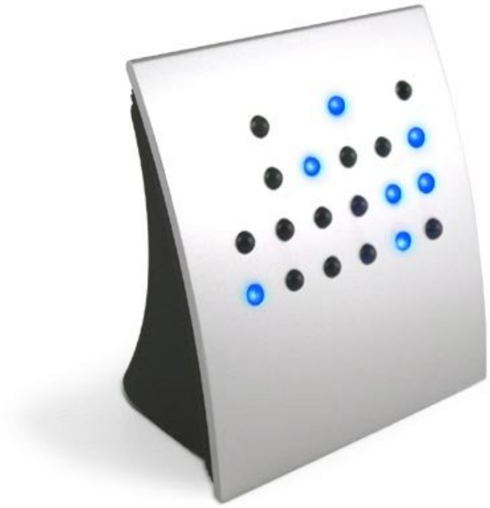
Binary Clock in MIDletPascal



```
{
	BinaryClock.mpsrc

	Created on Thursday, July 14, 2005
	Created with MIDletPascal (http://www.midletpascal.com)

	Author: Krisztian KUSICS
	E-mail: z n o s @ f r e e m a i l . h u
	Web:    http://inf.nyme.hu/~kusicsk/
}


program BinaryClock;

var
	images : array[0..1] of image;
	time : integer;
	xpos : integer;
	ypos : integer;


procedure drawBits(y: integer; binStr: string);
var
	i : integer;
	x : integer;
begin
	x := xpos;
	for i := 0 to ( length(binStr) - 1 ) do
	begin
		drawImage(images[stringToInteger(getChar(binStr, i))], x, y);
		x := x + 11;
	end;
end;


function integerToBinaryString(val: integer) : string;
var
	binaryString : string;
	index : integer;
	bit : integer;
begin
	binaryString := '00000000';
	index := 7;
	val := val * 2;
	while ( (val div 2) > 0 ) do
```

```
        begin
                val := val div 2;
                bit := val mod 2;
                binaryString := setChar(binaryString, getChar(integerToString(bit), 0), index);
                index := index - 1;
        end;
        integerToBinaryString := binaryString;
end;


begin
        images[0] := loadImage('/white.png');
        images[1] := loadImage('/red.png');

        xpos := (getWidth div 2) - 44;
        ypos := (getHeight div 2) - 16;

        repeat
                //setColor(255, 255, 255);
                //fillRect(0, 0, getWidth, getHeight);
                time := getCurrentTime;

                drawBits(ypos, integerToBinaryString(getHour(time)));
                drawBits(ypos + 11, integerToBinaryString(getMinute(time)));
                drawBits(ypos + 22, integerToBinaryString(getSecond(time)));

                repaint;
                delay(100);

                while isMidletPaused do
                begin
                        delay(100);
                end;

        until getKeyPressed <> KE_NONE;

        { halt; }

end.

{ EOF }
```