

Starburst in ACTION!#

MODULE

; Starburst 1.0 By Dan Rhea

; Modified from my Microsoft Basic
; version 7/85

DEFINE YES="1", ; Define logical flags
NO ="0"

CARD tc01, ; Scratch Card 1

console=[53279], ; Console Keys
option =[3], ; OPTION
select =[5], ; SELECT
start =[6], ; START
lmargin=[82], ; left margin
cursor =[752], ; cursor control
attract=[77] ; attract mode

INT ARRAY x(8), ; 8 possible x coordinates
y(8) ; 8 possible y coordinates

CARD xaxis ; X axis for plotting
BYTE yaxis ; Y axis for plotting

BYTE xmax=[48], ; Maximum x coordinate
ymax=[48], ; Maximum y coordinate
xtot=[95], ; Reflected maximum for xmax
ytot=[95], ; Reflected maximum for ymax
bias=[32], ; X coordinate offset from 0 (centers output)
lcol=[0], ; last color selected
mseg=[50], ; Maximum segment length (75 with 50 default)
adjx, ; X coordinate adjustment
adjy, ; Y coordinate adjustment
dirc, ; Direction of plot travel (1 to 8)
colr, ; Color of segment (1 to 3)
segl, ; Length of segment (1 to mseg)
move, ; movement counter
spot, ; plot counter/pointer
wrap=[YES], ; Wraparound flag
glue=[YES], ; Connected segments flag
tb01, ; Scratch byte 1
voic, ; Voice
pitc, ; Pitch
dist, ; Distortion
volu, ; Volume
ckey ; Console key

CHAR ansr ; Prompt answer

PROC Intro ()

; Introduction to Starburst

PrintE ("}")

Graphics (18)

```
Position (0,1)
PrintDE (6," OoOoOoOoOoOoOoOo")
PrintDE (6," o                o")
PrintDE (6," O StArBuRsT 1.0 O")
PrintDE (6," o                o")
PrintDE (6," O BY DAN rhea O")
PrintDE (6," o                o")
PrintDE (6," OoOoOoOoOoOoOoOo")
Position (0,10)
PrintDE (6," PRESS [start]")
```

```
DO
  FOR tc01=0 TO 3 ; Register select
  DO
    tb01=tc01
    colr=Rand(16)
    SetColor(tb01,colr,6)
  OD
  ckey=Peek(console)
  IF ckey <> start THEN
    FOR tc01=0 TO 10000 ; Delay
    DO
      ; Tarry a bit
    OD
  FI
UNTIL ckey = start
OD
```

RETURN

PROC Setup ()

; Set up drawing parameters

```
Graphics (0)
SndRst ()
Poke (lmargin,1)
PrintE ("-")
PrintE ("-----")
PrintE ("|                                |")
PrintE ("| Starburst 1.0 By Dan Rhea 07/15/85 |")
PrintE ("|                                |")
PrintE ("-----")
PrintE ("-----")
PrintE ("|This program will produce geometric |")
PrintE ("|patterns in Graphics Mode 7 using an|")
PrintE ("|8 way reflection algorithm. You can |")
PrintE ("|modify the type of patterns that are|")
PrintE ("|generated by altering the following:|")
PrintE ("|                                |")
PrintE ("| 1. Wraparound (line wrap or not)  |")
PrintE ("| 2. Connected Lines (does the next |")
PrintE ("|    line start where the last one  |")
PrintE ("|    completed)                      |")
PrintE ("| 3. Extent (Maximum line length in a|")
PrintE ("|    random direction)                |")
PrintE ("-----")
```

```
Poke (cursor,1) ; Cursor off
```

```
; Determine if wraparound is wanted
```

```
DO
```

```
Position (1,20)
```

```
PrintE ("-----")
```

```
PrintE ("|Wraparound Enable? (Y/N) :      |")
```

```
PrintE ("-----")
```

```
Position (28,21)
```

```
ansr=GetD(7)
```

```
IF ansr = 'Y OR ansr = 'y THEN
```

```
wrap = YES
```

```
tb01 = YES
```

```
ELSEIF ansr = 'N OR ansr = 'n THEN
```

```
wrap = NO
```

```
tb01 = YES
```

```
ELSE
```

```
tb01 = NO
```

```
FI
```

```
UNTIL tb01 = YES
```

```
OD
```

```
; Determine if connected lines are wanted
```

```
DO
```

```
Position (1,20)
```

```
PrintE ("-----")
```

```
PrintE ("|Connected lines required? (Y/N) :  |")
```

```
PrintE ("-----")
```

```
Position (35,21)
```

```
ansr=GetD(7)
```

```
IF ansr = 'Y OR ansr = 'y THEN
```

```
glue = YES
```

```
tb01 = YES
```

```
ELSEIF ansr = 'N OR ansr = 'n THEN
```

```
glue = NO
```

```
tb01 = YES
```

```
ELSE
```

```
tb01 = NO
```

```
FI
```

```
UNTIL tb01 = YES
```

```
OD
```

```
; Determine maximum line segment extent
```

```
DO
```

```
Position (1,20)
```

```
PrintE ("-----")
```

```
PrintE ("|Maximum segment length (1-75) :    |")
```

```
PrintE ("-----")
```

```
Position (33,21)
```

```
mseg=InputB()
```

```
IF mseg < 1 THEN
```

```
tb01 = NO
```

```
ELSEIF mseg > 75 THEN
```

```
tb01 = NO
```

```
ELSE
```

```

        tb01 = YES
    FI
UNTIL tb01 = YES
OD

; Give the user operating instructions during the draw mode

Position (1,20)
PrintE ("-----")
PrintE ("|START:Draw OPTION:Menu SELECT:Freeze|")
PrintE ("-----")
DO
    ckey=Peek(console)
UNTIL ckey = start
OD
DO
    ckey=Peek(console)
UNTIL ckey <> start
OD

Poke (cursor,0) ; Restore cursor

RETURN

PROC Getxy ()

; Set random X Y starting coordinates

x(1)=Rand(xmax) ; 0 to xmax-1
y(1)=Rand(ymax) ; 0 to ymax-1

RETURN

PROC DoClid ()

; Set Color, Length and Direction

dirc=Rand(8)      ; 0 to 7
DO
    colr=Rand(4)   ; 0 to 3
UNTIL colr <> lcol
OD
lcol=colr
segl=Rand(mseg)+1 ; 1 to mseg

RETURN

PROC Clamp ()

; Clamp the line or wrap it around as needed

IF wrap = YES THEN
    IF x(1) < 0 THEN
        adjx = xmax-1
    ELSEIF x(1) >= xmax THEN
        adjx = 0
    ELSE
        adjx = x(1)
    FI

```

```

x(1) = adjx
IF y(1) < 0 THEN
    adjy = ymax-1
ELSEIF y(1) >= ymax THEN
    adjy = 0
ELSE
    adjy = y(1)
FI
y(1) = adjy
ELSE
    IF x(1) < 0 THEN
        adjx = 0
    ELSEIF x(1) >= xmax THEN
        adjx = xmax-1
    ELSE
        adjx = x(1)
    FI
    x(1) = adjx
    IF y(1) < 0 THEN
        adjy = 0
    ELSEIF y(1) >= ymax THEN
        adjy = ymax-1
    ELSE
        adjy = y(1)
    FI
    y(1) = adjy
FI

```

RETURN

PROC Flect ()

; DO 8 way reflection

```

x(2) = xtot-x(1)
x(3) = x(2)
x(4) = x(1)
x(5) = y(1)
x(6) = xtot-x(5)
x(7) = x(6)
x(8) = x(5)
y(2) = y(1)
y(3) = ytot-y(1)
y(4) = y(3)
y(5) = x(1)
y(6) = y(5)
y(7) = ytot-y(6)
y(8) = y(7)

```

RETURN

PROC Paint ()

; Draw the sucker

```

FOR spot = 1 TO 8
DO
    xaxis=x(spot)+bias
    yaxis=y(spot)

```

```

        Plot (xaxis,yaxis)
    OD

RETURN

PROC Slide ()

; Move the guy in cell 1 in the desired direction

    IF dirc = 0 THEN
        x(1)==+1
    ELSEIF dirc = 1 THEN
        x(1)==+1
        y(1)==+1
    ELSEIF dirc = 2 THEN
        y(1)==+1
    ELSEIF dirc = 3 THEN
        x(1)==-1
        y(1)==+1
    ELSEIF dirc = 4 THEN
        x(1)==-1
    ELSEIF dirc = 5 THEN
        x(1)==-1
        y(1)==-1
    ELSEIF dirc = 6 THEN
        y(1)==-1
    ELSEIF dirc = 7 THEN
        x(1)==+1
        y(1)==-1
    ELSE
        ; Do Nothin Meng'
    FI

RETURN

PROC Noise ()

; Use screen data for sound

    BYTE base=[63]

    voic = colr
    pitc = x(1)+(colr*base)
    volu = y(1)/4
    dist = 10

    Sound(voic,pitc,dist,volu)

RETURN

PROC Main ()

    Intro ()

    DO
        Setup ()
        Getxy ()
        Graphics (23)
    DO

```

```

Poke(attract,0)
Docld ()
color=colr
FOR move = 1 TO segl
DO
    Clamp () ; Wrap/Nowrap
    Flect () ; Reflect
    Paint () ; Plot all 8
    Noise () ; Make some
    Slide () ; Move # 1
OD
IF glue = NO THEN
    Getxy ()
FI
ckey = Peek(console)
IF ckey = select THEN
    DO
        ckey = Peek(console)
        UNTIL ckey <> select
    OD
FI
IF ckey = start THEN
    DO
        ckey = Peek(console)
        UNTIL ckey <> start
    OD
    Graphics(23)
FI
UNTIL ckey = option
OD
OD
RETURN

```