

# B2- C Graphical Programming

---

B-MUL-151

## Scroller Rush

---

Effects by the thousands

v1.2



# Scroller Rush

Effects by the thousands

---

**binary name:** scroller  
**repository name:** scroller  
**repository rights:** ramassage-tek  
**language:** C  
**group size:** 3  
**compilation:** via Makefile, including re, clean and fclean rules

---



- Your repository must contain the totality of your source files, but no useless files (binary, temp files, obj files,...).
- All the bonus files (including a potential specific Makefile) should be in a directory named *bonus*.
- Error messages have to be written on the error output, and the program should then exit with the 84 error code (0 if there is no error).

Throughout this rush, you must create a demo that contains a set of mandatory effects, in addition to the ones you choose to include. The mandatory effect will result in malus if they are not complete !

The details of the points are available for each effect mentioned in the description.  
The more difficult the effects are, the more points they are worth.



Choose your effects smartly!

The grading scale will take the aesthetics of your demo into account. This criteria will be primarily based on the graphics-music synchronization, the inter-scenes transitions and the stability of a particular rhythm...

The quality of the graphic and sound resources is not in the grading scale, but they could potentially influence the examiner!



Mix up the effects!



## Mandatory effects

### Scrolling text

1pt(s)

Your demo should include a scrolling text.  
Once finished, your text should scroll again.

#### Conductor line

+0.25pt(s)

Your scrolling text shouldn't stop at moving from left to right, but should follow a diagonal trajectory.

#### Mobile conductor line

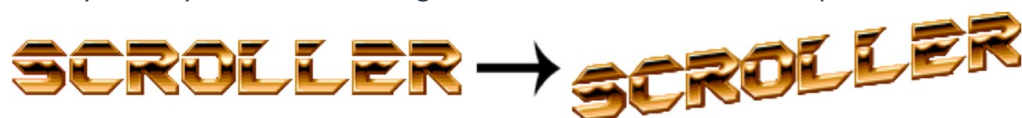
+0.25pt(s)

The conductor line isn't static: it evolves like a sine curve or triangular signal, for example.

#### Diagonal text

+0.5pt(s)

Not only should your text follow a diagonal conductor line, but each of its pixel column's letters should too.



### Parallax Scrolling

1pt(s)

Your demo should include a scrolling background.

This is an image that is repeated over and over on an axis, usually vertical or horizontal, and gives the impression of movement.

#### Multi-level scrolling

+0.5pt(s)

Your demo shouldn't include one scrolling image, but several of them, and all with different speeds. This difference in speed should give the impression of depth and the illusion of movement in a scene.

#### Scripted scrolling

+0.25pt(s)

Your scrolling shouldn't show a single similar infinite scene, but should show several different landscapes that can naturally come one after the other.

#### Multi-dimensional scrolling

+0.25pt(s)

Your scrolling evolves in different direction and not only vertically or horizontally.



## Tracker

1pt(s)

Your demo's musical background shouldn't be a single sound file, but the execution of a score passed as parameter or found in a configuration file. This tracker should be able to read at least one of the tracks given with this subject, like *beverly.bsf*.

It is not mandatory to exclusively use the tracker for the music, but your demo should at least include one. You must set up a sound manager.



man clock\_gettime

## Multi-track

+1pt(s)

Your tracker is able to read several tracks at the same time.

## Synthesizer

+0.5pt(s)

Your tracker is able to generate sounds instead of finding the ones in the files.

## Effects

+2.5pt(s)

Your tracker includes the "slide" and "vibrato" effects, which enables the sound to be distorted.

The slide allows you to go from one note to another by passing up all of the intermediaries.

The vibrato allows a sound to vibrate: the frequency evolves in sine waves with the original frequency as a carrier wave.



## Effects ?? la carte

### Starfield from the side

0.25pt(s)

*\*\* incompatible with the next effect \*\**

A scrolling starfield. This is a scroll in side-view.

### Frontal Starfield

0.5pt(s)

*\*\* incompatible with the previous effect \*\**

A starfield scrolls around you as though you were moving fast and looking towards your destination. The stars gradually become more and more visible as you approach them, therefore, they are invisible when they are too far away.

#### Rotating frontal Starfield

+0.25pt(s)

The starfield rotates around the Z axis. The rotation axis is located in the middle of the screen.

### Flames

0.25pt(s)

Flames.

#### Integrating the flames

+0.25pt(s)

The flames are included in scrolling or any other decor, and they are not in the background.

#### Wind

+0.25pt(s)

The flames flicker after wind blows, whose source you will determine.

#### Flames scrolling

+0.25pt(s)

Your flames scroll.

### Plasma

0.25pt(s)

A simple plasma made from mixed sine waves.

#### Plasma distortion

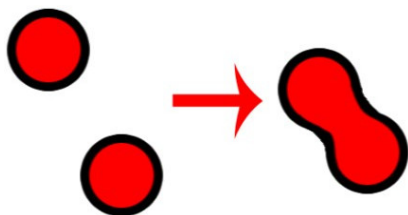
+0.25pt(s)

Your plasma's pattern isn't consistent.

### Plasma cells

1pt(s)

Balls of plasma move on the screen and appear to slightly merge together when they meet.



## Checkerboard mask

1pt(s)

A mask, in the form of a checkerboard, transforms the elements display that it covers. It's up to you to choose the transformation that is done. This transformation can potentially be another effect.



## Texture mask

+0.25pt(s)

Your mask is a texture.

## Image slider

2pt(s)

Your demo displays images by sliding them in the following way.



The sliding can also be carried out horizontally, and even merged with your scrolling text.

## Diagonal sliding

+0.25pt(s)

The image sliding is carried out diagonally.

## Image waves

0.5pt(s)

Your demo displays images by making them undulate:



## Multi-waves

+0.25pt(s)

The undulation can be sinusoidal, serrated or triangular.

## Multi-dimensions

+0.25pt(s)

The undulation is done simultaneously on two dimensions.

## Pseudo-fading by breaking down the elements

0.25pt(s)

Your demo includes a transition in which the images fade to black while breaking down; the elements in color are reduced to black depending on a threshold that evolves over time.

## Pseudo-Morphing

+0.25pt(s)

Your demo includes a transition that consists of passing one image to another through transparency.

## Mixing images

+0.25pt(s)

Different ways to mix existing images. Your demo includes at least two of these ways that aren't mentioned above. Here's an example of a mixed image:



## Text appearance

0.25pt(s)

Your demo makes text or small images aesthetically appear on the screen. Here's an example:



## Mobile shading

0.5pt(s)

Shading moves on the screen. It's a horizontal or vertical shaded strip, for example:



## Different sized shading

+0.5pt(s)

One of your mobile shadings is different in terms of size.



## Progressive shading

1pt(s)

The demo integrates shading whose colors change throughout. It's not their shape, but their color that changes.

## Transparent shading ??? Textured shading

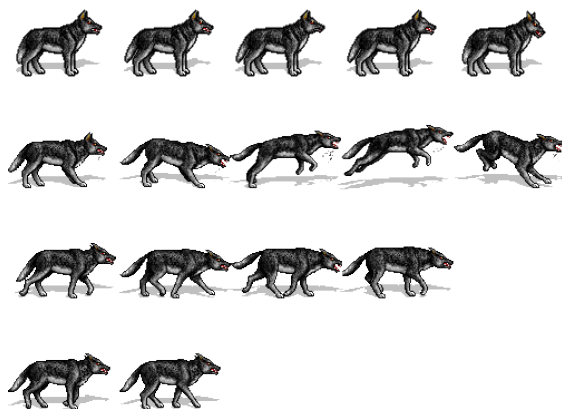
+0.5pt(s)

Your demo integrates shading whose content isn't a color but an image.

## Sprite animation

1pt(s)

Your demo shows an animated rendering with the help of a spriteboard. A spriteboard is available with the project description.



## Subwindows

0.5pt(s)

Different effects aren't rendered in the full-screen primary window, but in a small-screen subwindow.

### Mobile subwindow

+0.5pt(s)

The subwindows contain effects that move on-screen.

### Subwindow rotation

+0.25pt(s)

The subwindows carry out separate rotations in X or Y.

### Subwindow rotation with perspective

+0.25pt(s)

The subwindows can carry out separate X or Y rotations and visually come across a perspective-type distortion.

### Complete subwindow rotation

+0.5pt(s)

The subwindows can be turned around three axes, without any restrictions.



### Matrix 3D text

4pt(s)

Your demo integrates a rendered animation with the help of a spriteboard.  
A spriteboard is available with the project description.



### Thick 3D text

4pt(s)

Your demo displays a 3D text that carries out a rotation around the X or Y axis and integrates a pseudo-thickness. As follows:



### Full 3D object

6pt(s)

With the help of certain b3s files that are included in the library, such as cube.b3s, display your demo's objects in 3D, which will carry out rotations on 3 axes. The polygons must be complete and show a flat shadow from a bright spot that is located where you want.



Don't hesitate to learn about what a Z Buffer is in order to complete this effect.



### 3D background with focal distance

4pt(s)

Your multi-level scrolling integrates blurring depending on a variable point of view: if you wish to look "far away", your closer planes must be blurry. If you look between them, what is far and close should also be a little blurry. Your demo varies the perspective throughout.

### Normal mapping

4pt(s)

Your demo displays a 3D surface, like a giant tile on which the spectator is kept. On this giant tile, a relief will form according to the image. The lighter the color, the more elevated the relief. The normal\_map.png file that is provided for you is adapted to this treatment.

#### Surface rotation

+4pt(s)

Your surface in relief can carry out rotations on all three axes.

#### Texture

+1pt(s)

Your surface in relief can be textured with the help of an image.

---

## Authorized Functions

The entire LibC is authorized, except for the following functions:

- fork
- clone
- system
- popen
- exit
- abort

CSFML functions:

- all of Graphics module's functions
- all of System module's functions
- all of Window module's functions
- all of Audio module's functions