search...

igaither

(https://profile.intra.42.fr)

Create a new

profile.intra.42.fr)

Summary (/)

projects.intra.42.fr)

Latest topics (/topics/latest)

Tags

Filter tags

earning.intra.42.fr)

Starred tags

Ignored tags

forum.intra.42.fr)

ADM 7 (/Tags/12/Topics)
API (/Tags/593/Topics)
Archives (/Tags/526/Topics)
BDE 2 (/Tags/13/Topics)
Betting 1 (/Tags/132/Topics)
Blockchain 2 (/Tags/787/Topics)
Bocal 1 (/Tags/17/Topics)
Building 8 (/Tags/54/Topics)
C 39 (/Tags/2/Topics)
C++ 3 (/Tags/38/Topics)
CCMN 1 (/Tags/712/Topics)
C Exam - Intermediate (/Tags/674/Topics)
Cisco (/Tags/713/Topics)
Classifieds 2 (/Tags/14/Topics)
Clubs 4 (/Tags/11/Topics)
Coalitions (/Tags/840/Topics)
Electronics 5 (/Tags/43/Topics)
Events (/Tags/845/Topics)
Food Truck 3 (/Tags/20/Topics)
Frozen 1 (/Tags/678/Topics)
Game-Theory (/Tags/66/Topics)
General 32 (/Tags/8/Topics)
Global 48 (/Tags/322/Topics)
Graphics 11 (/Tags/27/Topics)
Housing 2 (/Tags/9/Topics)
Internship 8 (/Tags/42/Topics)
Intra 30 (/Tags/7/Topics)
KFS (/Tags/583/Topics)
Kyiv-ADM (/Tags/792/Topics)
Kyiv-Bocal (/Tags/798/Topics)
Kyiv-Internship (/Tags/793/Topics)
Kyiv-Lab (/Tags/794/Topics)
Kyiv-Translate (/Tags/797/Topics)
Lost And Found (/Tags/19/Topics)
Mathematics 4 (/Tags/728/Topics)

mpanies.intra.42.fr)

/meta.intra.42.fr)

/shop.intra.42.fr)

vance (/topics/19254/messages?sort=relevant) Time (/topics/19254/me

FdF Cookbook

Watch

Informations

FdF (/tags/26/topics) Graphics (/tags/27/topics) MiniLib

166
163
3
(https://profile.intra.42.fr/users/vbrazhni)

Created by vbrazhni (https://profile.intra.42.fr/users/vbrazhni) 2018-08-23

Altruist vbrazhni

Written in english (https://profile.intra.42.fr/users/vbrazhni)

Foreword 265 4 261

You can also read this information on my GitHub **wiki-pages**. Maybe it w

Link: https://github.com/VBrazhnik/FdF/wiki **494** (https://github.com,

How to handle mouse buttons and

4 2985 10

On `macOS`, if you want to handle mouse buttons and key presses and clos function instead of the other hook functions:

```
int mlx_hook(void *win_ptr, int x_event, int x_mas
```

6 2 1 1 1 1

To handle a key press

At the place of `int x_event` parametr use `2`.

At the place of `int (*funct)()` parameter you use the following functio

```
int key_press(int keycode, void *param)
```

To handle a key release

At the place of `int x_event` parametr use `3`.

At the place of `int (*funct)()` parameter you use the following functio

```
int key_release(int keycode, void *param)
```

To handle a mouse button press

At the place of `int x_event` parametr use `4`.

At the place of `int (*funct)()` parameter you use the following functio

```
int mouse_press(int button, int x, int y, void *pa
```

To handle a mouse button release

At the place of `int x_event` parametr use `5`.

At the place of `int (*funct)()` parameter you use the following functio

```
int mouse_release(int button, int x, int y, void *
```

To handle a mouse movement

At the place of `int x_event` parametr use `6`.

At the place of `int (*funct)()` parameter you use the following functio

```
int mouse_move(int x, int y, void *param)
```

To handle an expose event

At the place of `int x_event` parametr use `12`.

3 6 93

Data Definitions for libX11 **547** (https://refspecs.linuxfoundation.org/LSB_1.3.0/gLSB/gddefs.html) refspecs.linuxfoundation.org

Pages · VBrazhnik/FdF Wiki · GitHub **495** (https://github.com/VBrazhnik/FdF/wiki) github.com

Bresenham's line algorithm - Wikipedia **309** (https://en.wikipedia.org/wiki/Bresenham%27s_line_alg en.wikipedia.org

Rotation matrix - Wikipedia **295** (https://en.wikipedia.org/wiki/Rotation_matrix) en.wikipedia.org

Xiaolin Wu's line algorithm - Wikipedia **255** (https://en.wikipedia.org/wiki/Xiaolin_Wu%27s_line_al en.wikipedia.org

Компьютерная графика **231** (https://grafika.me/node/82) grafika.me

Isometric 2:1 Projections: Isometric Infographic Vectors - Vectips **189** (http://vectips.com/tutorials/isometric-infographic-vectors/) vectips.com

Kirupa.com : Isometric Transformation **177** (https://www.kirupa.com/developer/actionscript/isome

How do I calculate color gradients? - Graphic Design Stack Exchange **139** (https://graphicdesign.stackexchange.com/a/109026) graphicdesign.stackexchange.com

Geometry | How Does Matrix Work? Part 1 **138** (http://www.scratchapixel.com/lessons/mathematics-physics-for-computer-graphics/geometry/how-does-matrix-work-part-1) scratchapixel.com

Tags

FdF (/tags/26/topics) Graphics (/tags/27/topics) MiniLibX (/tags/716/topics)

At the place of `int (*funct)()` parameter you use the following functi

```
int expose(void *param)
```

To handle a red button (`X` button) press

At the place of `int x_event` parametr use `17` .

At the place of `int (*funct)()` parameter you use the following functi

```
int close(void *param)
```

Complete `int close(void *param)` function:

```
int close(void *param)
{
    (void)param;
    exit(0);
}
```

Tip:

`x_mask` is ignored on macOS. But if you want that your `FdF` will have cc

Key codes

Mouse button codes

- Left button — 1
- Right button — 2
- Third (Middle) button — 3
- Scroll Up — 4
- Scroll Down — 5
- Scroll Left — 6
- Scroll Right — 7

Masks

You can find values of `x_mask` here **544** (https://refspecs.linuxfoundati

Tip:

`x_mask` for `int close(void *param)` is `(1L << 17)` .

Created by vbrazhni (https://profile.intra.42.fr/users/vbrazhni) **2018-08-23 12:29** — 1.7K word
Posted in cursus **42** — Written in **English**

19
19
0
(https://profile.intra.42.fr/users/vbrazhni)

Altruist vbrazhni
(https://profile.intra.42.fr/users/vbrazhni)

How to draw a line?

To draw a line you can use Bresenham's line algorithm **281** (https://en.wi solution) or Xiaolin Wu's line algorithm **236** (https://en.wikipedia.org/wi solution which will produced more beautiful result).

21
21
0
(https://profile.intra.42.fr/users/vbrazhni)

Altruist vbrazhni
(https://profile.intra.42.fr/users/vbrazhni)

How to create linear gradient?

Here we will consider how to find the color between any two color points th

First of all we need to find current point position between two points with kn percentages.

The following function will help you find this value:

```
double percent(int start, int end, int current)
{
    double placement;
    double distance;

    placement = current - start;
    distance = end - start;
    return ((distance == 0) ? 1.0 : (placement / d
}
```

You can calculate this value depending on which delta value is bigger. Delt values.

Part of code:

```
// ...
    double percentage;

    if (delta.x > delta.y)
        percentage = percent(start.x, end.x, curre
    else
        percentage = percent(start.y, end.y, curre
// ...
```

Then for creating each light (**R**ed, **G**reen, **B**lue) we need to get light from st need to get new color by union red, green and blue light.

Part of code:

```
// ...
    int red;
    int green;
    int blue;

    // Get percentage

    red = get_light((start.color >> 16) & 0xFF, (e
    green = get_light((start.color >> 8) & 0xFF, (
    blue = get_light(start.color & 0xFF, end.color
    return ((red << 16) | (green << 8) | blue);
```

```
int get_light(int start, int end, double percentag
{
    return ((int)((1 - percentage) * start + perce
}
```

Complete code:

```
int get_light(int start, int end, double percentag
{
    return ((int)((1 - percentage) * start + perce
}

int get_color(t_point current, t_point start, t_po
{
    int     red;
    int     green;
    int     blue;
    double  percentage;

    if (current.color == end.color)
        return (current.color);
    if (delta.x > delta.y)
        percentage = percent(start.x, end.x, curre
    else
        percentage = percent(start.y, end.y, curre
    red = get_light((start.color >> 16) & 0xFF, (e
    green = get_light((start.color >> 8) & 0xFF, (
    blue = get_light(start.color & 0xFF, end.color
    return ((red << 16) | (green << 8) | blue);
}
```

Basic information was found here 126 (https://graphicdesign.stackexcha

Color for pixel

Everything is easy if you decided to use the following function:

```
int mlx_pixel_put(void *mlx_ptr, void *win_ptr, in
```

In this case, the order of lights is standard:

0 R G B
8 bits8 bits8 bits8 bits
As you can see that the first byte is filled with zeros. It means that the alpha

You can find this information in `mlx_pixel_put` man file.

Also, this information is actual for color parameter in the function which disp

```
int mlx_string_put(void *mlx_ptr, void *win_ptr, i
```

But if you decided to use an image, you will face with more complicated us

You will work with the following three functions:

```
void *mlx_new_image(void *mlx_ptr, int width, int
```

```
char *mlx_get_data_addr(void *img_ptr, int *bits_p
```

```
int mlx_put_image_to_window(void *mlx_ptr, void *w
```

And the most interesting is the second function with such parameters as `bit`

> **What is bits per pixel or bit-depth value?**
>
> The number of bits used to define a pixel's color shade is
> as 24-bit color. Some new color display systems offer a
> alpha channel, is used for control and special effects infc

For `macOS` value of `bits_per_pixel` is constant. You can find the follo

```
#define UNIQ_BPP 4
```

```
// assume here 32bpp little endian

char *mlx_get_data_addr(mlx_img_list_t *img_ptr, i
 *endian)
{
    *bits_per_pixel = UNIQ_BPP * 8;
    *size_line = img_ptr->width * UNIQ_BPP;
    // ...
}
```

If you decided to support only `macOS` , you don't need to worry about the bytes (32 bits).

`endian` is the most important parameter that we have to consider.

For `macOS` its value is `0` , which means `little endian` .

Information about `endian` value you can also find in source files of minilib

```
/*
** endian : 0 = sever X is little endian, 1 = big
** endian : useless on macos, client and graphical
*/
```

```
// assume here 32bpp little endian

char *mlx_get_data_addr(mlx_img_list_t *img_ptr, i
 *endian)
{
    // ...
    *endian = 0; // little endian for now on mac-i
    // ...
}
```

Big-endian and little-endian are the formats of ordering bytes.

Big-endian is the format that we used to know as **normal**.

Little-endian order is **reversed**.

For color these two formats look like:

Byte number0 1 2 3
Big endian 0 R G B
Little endian B G R 0
So in the case of little-endian format, you have to use reversed order of col

```
// ...
    int i;

    i = (x * fdf->bits_per_pixel / 8) + (y * fdf->
    fdf->data_addr[i] = color; // B - Blue
    fdf->data_addr[++i] = color >> 8; // G - Green
    fdf->data_addr[++i] = color >> 16; // R - Red
    fdf->data_addr[++i] = 0; // Alpha channel
// ...
```

22
21
1
(https://profile.intra.42.fr/users/vbrazhni)

Altruist vbrazhni
(https://profile.intra.42.fr/users/vbrazhni)

How to rotate figure in 3D?

If you want to rotate a vector you should construct what is known as a rotati
(https://en.wikipedia.org/wiki/Rotation_matrix).

X-Axis Rotation

$$R_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & -sin(\theta) \\ 0 & sin(\theta) & cos(\theta) \end{bmatrix}$$

After the transformations, we will get the formulas:

$$x' = x;$$
$$y' = y * cos(\theta) + z * sin(\theta);$$
$$z' = -y * sin(\theta) + z * cos(\theta);$$

Y-Axis Rotation

$$R_y(\theta) = \begin{bmatrix} cos(\theta) & 0 & sin(\theta) \\ 0 & 1 & 0 \\ -sin(\theta) & 0 & cos(\theta) \end{bmatrix}$$

After the transformations, we will get the formulas:

$$x' = x * cos(\theta) + z * sin(\theta);$$
$$y' = y;$$
$$z' = -x * sin(\theta) + z * cos(\theta);$$

Z-Axis Rotation

$$R_z(\theta) = \begin{bmatrix} cos(\theta) & -sin(\theta) & 0 \\ sin(\theta) & cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

After the transformations, we will get the formulas:

$$x' = x * cos(\theta) - y * sin(\theta);$$
$$y' = x * sin(\theta) + y * cos(\theta);$$
$$z' = z;$$

Source of information 227 (http://grafika.me/node/82) (Russian)

20
20
0
(https://profile.intra.42.fr/users/vbrazhni)

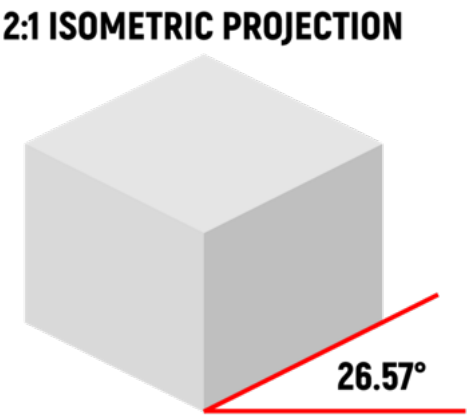
Altruist vbrazhni
(https://profile.intra.42.fr/users/vbrazhni)

How to perform isometric transform

There are **"true"** isometric projection and **2:1** isometric projection.

True isometric projection uses a 30° angle (0.523599 rad).

2:1 isometric projection uses a 26.57° angle (0.46373398 rad).



Source of information 179 (http://vectips.com/tutorials/isometric-infogr

Code for transforming:

```
static void iso(int *x, int *y, int z)
{
    int previous_x;
    int previous_y;

    previous_x = *x;
    previous_y = *y;
    *x = (previous_x - previous_y) * cos(0.523599)
    *y = -z + (previous_x + previous_y) * sin(0.52
}

t_point project(t_point p, t_fdf *fdf)
{
// ...
    if (fdf->camera->projection == ISO)
        iso(&p.x, &p.y, p.z);
// ...
}
```

Source of information 167 (https://www.kirupa.com/developer/actions

Userspace_digressions (/Tags/753/Top
Walking Marvin 1 (/Tags/647/Topics)
War 2 (/Tags/618/Topics)
Web Initiation (/Tags/524/Topics)
Wildcard (/Tags/827/Topics)
Wolf3d 4 (/Tags/30/Topics)
Woody Woodpacker (/Tags/549/Topics)
XV (/Tags/535/Topics)
Yellow Brick Road 1 (/Tags/784/Topics)
Zappy 1 (/Tags/107/Topics)

7 0 7

(https://profile.intra.42.fr/users/mtaylor)

mtaylor (https://profile.intra.42.fr/users/mtaylc

This transformation is pretty good, but it results in a "backv says "42", but the text is reversed.)

To fix this, invert the + and -:

```
new_x = (x + y) * cos(angle);
new_y = (x - y) * sin(angle) - z;
```

2 0 2

(https://profile.intra.42.fr/users/vbrazhni)

Altruist vbrazhni (https://profile.intra.42.fr/users

It depends only on your realization of reading and storag correct image after projection, someone will get a reverse

You are completely right that it is simply to fix. You can cor (changing sign before x, y or z) to reverse axis direction. A

The format of formulas, that are listed here, is the best for r perfect result.

7 0 7

(https://profile.intra.42.fr/users/mtaylor)

mtaylor (https://profile.intra.42.fr/users/mtaylc

Slanting right:

```
new_x = (x + y) * cos(angle);
new_y = (x - y) * sin(angle) - z;
```

Slanting left:

```
new_x = (x - y) * -cos(angle);
new_y = ((x + y) * sin(angle)) - z;
```

Mirror image slanting right:

```
new_x = (x - y) * cos(angle);
new_y = ((x + y) * sin(angle)) - z;
```

Mirror image slanting left:

```
new_x = (x + y) * -cos(angle);
new_y = ((x - y) * sin(angle)) - z;
```

0 0 0

(https://profile.intra.42.fr/users/cfargere)

cfargere

(https://profile.intra.42.fr/users/cfargere)

Interestingly, the rotation matrix shown here are reversed from the scratchpi (http://www.scratchapixel.com/lessons/mathematics-physics-for-comput

0 0 0

(https://profile.intra.42.fr/users/cyuriko)

cyuriko

(https://profile.intra.42.fr/users/cyuriko)

Most helpful, thanks!

0
0
0

bshara
(https://profile.intra.42.fr/users/bshara)

(https://profile.intra.42.fr/users/bshara)
thanks😄

0
0
0

jmacgyve
(https://profile.intra.42.fr/users/jmacgyve)

(https://profile.intra.42.fr/users/jmacgyve)
th

1
1
0

samymone
(https://profile.intra.42.fr/users/samymone)

(https://profile.intra.42.fr/users/samymone)
very helpful, thanks!)

Loading next messages...

Type some words...