

sprites4curses

0.1.4

Generated by Doxygen 1.9.6

1 Documentation	1
2 palette.gpl	3
3 sprites4curses	5
3.1 sprites.py	5
3.2 sheet_converter.py	5
3.3 png_resize.py	5
3.4 palette.gpl	5
3.5 animate.c and animate.h	6
3.6 demo.c	6
3.7 Usage	6
3.7.0.1 This overwrites the source pngs, so be careful.	6
3.7.1 The demo program is meant to show how to correctly call <code>animate_file()</code> from <code>animate.h</code> .	6
4 Namespace Index	7
4.1 Namespace List	7
5 File Index	9
5.1 File List	9
6 Namespace Documentation	11
6.1 png_resize Namespace Reference	11
6.1.1 Detailed Description	11
6.1.2 Function Documentation	11
6.1.2.1 <code>main()</code>	11
6.1.2.2 <code>resize_sprites()</code>	11
6.1.2.3 <code>usage()</code>	12
6.2 sheet_converter Namespace Reference	12
6.2.1 Detailed Description	12
6.2.2 Function Documentation	12
6.2.2.1 <code>convert_spritesheet()</code>	12
6.2.2.2 <code>main()</code>	13
6.2.2.3 <code>usage()</code>	13
6.3 sprites Namespace Reference	13
6.3.1 Detailed Description	14
6.3.2 Function Documentation	14
6.3.2.1 <code>color_distance()</code>	14
6.3.2.2 <code>convert_sprite()</code>	14
6.3.2.3 <code>create_palette()</code>	14
6.3.2.4 <code>main()</code>	15
6.3.2.5 <code>print_converted_sprites()</code>	15
6.3.2.6 <code>usage()</code>	15

7 File Documentation	17
7.1 sprites4curses/animate.c File Reference	17
7.1.1 Function Documentation	17
7.1.1.1 animate_file()	17
7.1.1.2 init_s4c_color_pairs()	18
7.1.1.3 load_sprites()	18
7.2 sprites4curses/animate.h File Reference	18
7.2.1 Macro Definition Documentation	18
7.2.1.1 BLACK	19
7.2.1.2 BLUE	19
7.2.1.3 CYAN	19
7.2.1.4 GREEN	19
7.2.1.5 MAGENTA	19
7.2.1.6 MAXCOLS	19
7.2.1.7 MAXFRAMES	19
7.2.1.8 MAXROWS	20
7.2.1.9 RED	20
7.2.1.10 WHITE	20
7.2.1.11 YELLOW	20
7.2.2 Function Documentation	20
7.2.2.1 animate_file()	20
7.2.2.2 init_s4c_color_pairs()	20
7.2.2.3 load_sprites()	21
7.3 animate.h	21
7.4 sprites4curses/demo.c File Reference	21
7.4.1 Macro Definition Documentation	22
7.4.1.1 DEMOCOLS	22
7.4.1.2 DEMOFRAMES	22
7.4.1.3 DEMOFRAMETIME	22
7.4.1.4 DEMOROWS	22
7.4.2 Function Documentation	22
7.4.2.1 demo()	22
7.4.2.2 main()	23
7.4.2.3 usage()	23
7.5 sprites4curses/documentation/README.md File Reference	23
7.6 sprites4curses/README.md File Reference	23
7.7 sprites4curses/palette-README.md File Reference	23
7.8 sprites4curses/png_resize.py File Reference	23
7.8.1 Detailed Description	23
7.8.2 Description	24
7.8.3 Libraries/Moodules	24
7.8.4 Notes	24

7.8.5 TODO	24
7.8.6 Author(s)	24
7.9 sprites4curses/sheet_converter.py File Reference	24
7.9.1 Detailed Description	25
7.9.2 Description	25
7.9.3 Libraries/Moodules	25
7.9.4 Notes	25
7.9.5 TODO	25
7.9.6 Author(s)	25
7.10 sprites4curses/sprites.py File Reference	26
7.10.1 Detailed Description	26
7.10.2 Description	26
7.10.3 Libraries/Moodules	26
7.10.4 Notes	27
7.10.5 TODO	27
7.10.6 Author(s)	27
Index	29

Chapter 1

Documentation

These folders contain documentation on the project, generated using Doxygen's latex output to pdf.

Chapter 2

palette.gpl

If your image does not have a palette of 256 colors, you can convert it to 8-bit indexed color mode with a custom palette in GIMP.

- + Open the image in GIMP.
- + Select Image > Mode > Indexed.
- + In the Indexed Color Conversion dialog, choose "Generate optimum palette" as the conversion type.
- + Under the "Maximum number of colors" option, enter "256".
- + Check the "Use custom palette" checkbox.
- + Click the "Edit palette" button.
- + In the Palette Editor dialog, click the "Import Palette" button.
- + Select "From file" and choose the palette file (palette.gpl).
- + Click "OK" to close the Palette Editor dialog.
- + Click "Convert" in the Indexed Color Conversion dialog to convert the image to indexed color mode with the c
- + Export the image in PNG format.

Chapter 3

sprites4curses

A library of scripts to deal with sprites in ncurses.

3.1 sprites.py

This is a python script that converts PNG's to a char representation. The output text should be a valid C declaration for a 3D char array.

It expects as arguments a directory with the images to convert. There's a dependency on Pillow to do the image conversion.

3.2 sheet_converter.py

This is a python script that converts a single PNG spritesheet to a char representation. The output text should be a valid C declaration for a 3D char array.

It expects as arguments the spritesheet file name, the sprite width, the sprite height, the thickness of the separator between sprites, and the start coordinate or the first sprite's left corner. There's a dependency on Pillow to do the image conversion.

3.3 png_resize.py

This is a python script that resizes PNG's to a desired size.

It expects as arguments a directory with the images to resize, and two ints for width and height of the resulting PNGs. There's a dependency on Pillow to do the image conversion.

3.4 palette.gpl

This is a GIMP palette file, useful for exporting PNG with the correct color alignment. Info on how to use it are in the palette-Readme.md file.

3.5 `animate.c` and `animate.h`

This is a C program to display an animation read from a formatted text file. They offer the function `animate_file()` useful to animate in a initialised WINDOW. The format expected is compatible with `sprites.py` specs.

3.6 `demo.c`

This is a demo program directly using the `demo()` function from the library.

3.7 Usage

To use the python scripts you need to install Pillow:

3.7.0.0.1 `<tt>pip install Pillow</tt>`

- To run the sprites script and redirect output on "file.txt", give a directory to get the png's from:

3.7.0.0.2 File names in the directory should follow a `imageX.png`, `imageX+1.png` pattern.

3.7.0.0.3 `<tt>python sprites.py <directory> > file.txt</tt>`

- To run the sheet converter script and redirect output on "file.txt", give all required arguments:

3.7.0.0.4 `<tt>python sheet_converter.py <sheet file> <sprite width> <sprite height> <separator thickness> <starting coordinate> > file.txt</tt>`

- To run the png resize script, give all required arguments:

3.7.0.0.5 `<tt>python png_resize.py <sprites directory> <sprite width> <sprite height></tt>`

3.7.0.1 This overwrites the source pngs, so be careful.

3.7.1 The demo program is meant to show how to correctly call `animate_file()` from `animate.h`.

- To run the C demo program, you do:

3.7.1.0.1 The demo is meant to run with the provided file.

3.7.1.0.2 `<tt>make; ./demo demofile.txt</tt>`

- To be fancy you can use process substitution in bash to give the python output directly as an argument:

3.7.1.0.3 `<tt>make; ./demo <(python sprites.py <directory>)</tt>`

3.7.1.0.4 Possible animation glitches if the frame rate is too high, add in-between frames as needed.

Chapter 4

Namespace Index

4.1 Namespace List

Here is a list of all namespaces with brief descriptions:

png_resize	Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel .	11
sheet_converter	Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel .	12
sprites	Program that parses pngs from a passed directory, to encode their color to a char per pixel . .	13

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

sprites4curses/ animate.c	17
sprites4curses/ animate.h	18
sprites4curses/ demo.c	21
sprites4curses/ png_resize.py	
Program that resizes pngs to a desired size and overwrites them	23
sprites4curses/ sheet_converter.py	
Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel .	24
sprites4curses/ sprites.py	
Program that parses pngs from a passed directory, to encode their color to a char per pixel . .	26

Chapter 6

Namespace Documentation

6.1 png_resize Namespace Reference

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

Functions

- def `usage` ()
Prints correct invocation.
- def `resize_sprites` (directory, targetSizeX, targetSizeY)
Resizes all png files in the passed directory to the specified size.
- def `main` (argv)
Main program entry.

6.1.1 Detailed Description

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

6.1.2 Function Documentation

6.1.2.1 main()

```
def png_resize.main (
    argv )
```

Main program entry.

6.1.2.2 resize_sprites()

```
def png_resize.resize_sprites (
    directory,
    targetSizeX,
    targetSizeY )
```

Resizes all png files in the passed directory to the specified size.

Parameters

<i>directory</i>	The input directory with the pngs.
<i>targetSizeX</i>	The target width.
<i>targetSizeY</i>	The target height.

6.1.2.3 usage()

```
def png_resize.usage ( )
```

Prints correct invocation.

6.2 sheet_converter Namespace Reference

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

Functions

- def [usage](#) ()
Prints correct invocation.
- def [convert_spritesheet](#) (filename, spriteSizeX, spriteSizeY, separatorSize, startCoords)
Converts a spritesheet to a 3D char array representation of pixel color and then prints it with the needed brackets and commas.
- def [main](#) (argv)
Main program entry.

6.2.1 Detailed Description

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

6.2.2 Function Documentation**6.2.2.1 convert_spritesheet()**

```
def sheet_converter.convert_spritesheet (
    filename,
    spriteSizeX,
    spriteSizeY,
    separatorSize,
    startCoords )
```

Converts a spritesheet to a 3D char array representation of pixel color and then prints it with the needed brackets and commas.

Parameters

<i>filename</i>	The input spritesheet file.
<i>spriteSizeX</i>	The sprite width.
<i>spriteSizeY</i>	The sprite height.
<i>separatorSize</i>	Thickness of separator pixels.
<i>startCoords</i>	Coords (a,a) of left corner of first sprite.

6.2.2.2 main()

```
def sheet_converter.main (
    argv )
```

Main program entry.

6.2.2.3 usage()

```
def sheet_converter.usage ( )
```

Prints correct invocation.

6.3 sprites Namespace Reference

Program that parses pngs from a passed directory, to encode their color to a char per pixel.

Functions

- def [usage](#) ()
Prints correct invocation.
- def [color_distance](#) (c1, c2)
Calculates the distance in color between two rgb tuples.
- def [create_palette](#) (colors)
Creates a palette by adding the colors passed as an array, then pads the palette.
- def [convert_sprite](#) (file)
Takes a image file and converts each pixel to a char representation of its color (closest match to CHAR_MAP).
- def [print_converted_sprites](#) (direc)
Takes a directory containing image file and calls convert_sprite on each one.
- def [main](#) (argv)
Main program entry.

6.3.1 Detailed Description

Program that parses pngs from a passed directory, to encode their color to a char per pixel.

6.3.2 Function Documentation

6.3.2.1 `color_distance()`

```
def sprites.color_distance (
    c1,
    c2 )
```

Calculates the distance in color between two rgb tuples.

Parameters

<i>c1</i>	The first input color to measure.
<i>c2</i>	The second input color to measure.

Returns

The color distance between the two.

6.3.2.2 `convert_sprite()`

```
def sprites.convert_sprite (
    file )
```

Takes a image file and converts each pixel to a char representation of its color (closest match to CHAR_MAP).

Parameters

<i>file</i>	The image file to convert.
-------------	----------------------------

Returns

The converted sprite as a char matrix.

6.3.2.3 `create_palette()`

```
def sprites.create_palette (
    colors )
```

Creates a palette by adding the colors passed as an array, then pads the palette.

Parameters

<i>colors</i>	The colors to put in the palette.
---------------	-----------------------------------

Returns

The filled and padded palette.

6.3.2.4 main()

```
def sprites.main (
    argv )
```

Main program entry.

6.3.2.5 print_converted_sprites()

```
def sprites.print_converted_sprites (
    direc )
```

Takes a directory containing image file and calls convert_sprite on each one.

Then it outputs all the converted sprites to stdout, including the necessary brackets to have a valid C array declaration.

Parameters

<i>direc</i>	The directory of image files to convert and print.
--------------	--

6.3.2.6 usage()

```
def sprites.usage ( )
```

Prints correct invocation.

Chapter 7

File Documentation

7.1 sprites4curses/animate.c File Reference

```
#include <ncurses.h>
#include <string.h>
#include <ctype.h>
#include <stdlib.h>
#include "animate.h"
```

Functions

- void [load_sprites](#) (char sprites[[MAXFRAMES](#)][[MAXROWS](#)][[MAXCOLS](#)], const char *filename, int rows, int columns)
- void [init_s4c_color_pairs](#) ()
- void [animate_file](#) (WINDOW *w, char *filename, int repetitions, int frametime, int num_frames, int frameheight, int framewidth)

7.1.1 Function Documentation

7.1.1.1 [animate_file\(\)](#)

```
void animate_file (
    WINDOW * w,
    char * filename,
    int repetitions,
    int frametime,
    int num_frames,
    int frameheight,
    int framewidth )
```

7.1.1.2 `init_s4c_color_pairs()`

```
void init_s4c_color_pairs ( )
```

7.1.1.3 `load_sprites()`

```
void load_sprites (
    char sprites[MAXFRAMES][MAXROWS][MAXCOLS],
    const char * filename,
    int rows,
    int columns )
```

7.2 `sprites4curses/animate.h` File Reference

```
#include <ncurses.h>
```

Macros

- `#define RED 1`
- `#define YELLOW 2`
- `#define GREEN 3`
- `#define CYAN 4`
- `#define BLUE 5`
- `#define MAGENTA 6`
- `#define BLACK 7`
- `#define WHITE 8`
- `#define MAXFRAMES 121`
Defines the maximum number of sprites.
- `#define MAXROWS 26`
Defines the maximum number of rows per sprite.
- `#define MAXCOLS 84`
Defines the maximum number of columns per sprite.

Functions

- void `load_sprites` (char sprites[MAXFRAMES][MAXROWS][MAXCOLS], const char *filename, int rows, int columns)
- void `init_s4c_color_pairs` ()
- void `animate_file` (WINDOW *w, char *filename, int repetitions, int frametime, int num_frames, int frame-height, int framewidth)

7.2.1 Macro Definition Documentation

7.2.1.1 BLACK

```
#define BLACK 7
```

7.2.1.2 BLUE

```
#define BLUE 5
```

7.2.1.3 CYAN

```
#define CYAN 4
```

7.2.1.4 GREEN

```
#define GREEN 3
```

7.2.1.5 MAGENTA

```
#define MAGENTA 6
```

7.2.1.6 MAXCOLS

```
#define MAXCOLS 84
```

Defines the maximum number of columns per sprite.

7.2.1.7 MAXFRAMES

```
#define MAXFRAMES 121
```

Defines the maximum number of sprites.

7.2.1.8 MAXROWS

```
#define MAXROWS 26
```

Defines the maximum number of rows per sprite.

7.2.1.9 RED

```
#define RED 1
```

7.2.1.10 WHITE

```
#define WHITE 8
```

7.2.1.11 YELLOW

```
#define YELLOW 2
```

7.2.2 Function Documentation

7.2.2.1 animate_file()

```
void animate_file (
    WINDOW * w,
    char * filename,
    int repetitions,
    int frametime,
    int num_frames,
    int frameheight,
    int framewidth )
```

7.2.2.2 init_s4c_color_pairs()

```
void init_s4c_color_pairs ( )
```

7.2.2.3 load_sprites()

```
void load_sprites (
    char sprites[MAXFRAMES][MAXROWS][MAXCOLS],
    const char * filename,
    int rows,
    int columns )
```

7.3 animate.h

[Go to the documentation of this file.](#)

```
1 #include <ncurses.h>
2 //These define the colors for init_pair() without an order
3 #define RED 1
4 #define YELLOW 2
5 #define GREEN 3
6 #define CYAN 4
7 #define BLUE 5
8 #define MAGENTA 6
9 #define BLACK 7
10 #define WHITE 8
11
12 #define MAXFRAMES 121
13 #define MAXROWS 26
14 #define MAXCOLS 84
16 static void print_spriteline(WINDOW* w, char* line, int line_num);
17 static char *trim(char *str);
18 void load_sprites(char sprites[MAXFRAMES][MAXROWS][MAXCOLS], const char *filename, int rows, int
    columns);
19 void init_s4c_color_pairs();
20 void animate_file(WINDOW* w, char* filename, int repetitions, int frametime, int num_frames, int
    frameheight, int framewidth);
```

7.4 sprites4curses/demo.c File Reference

```
#include <unistd.h>
#include <stdlib.h>
#include <locale.h>
#include "animate.h"
```

Macros

- `#define DEMOFRAMES 30`
Defines the number of sprites in the demo.
- `#define DEMOROWS 18`
Defines the maximum number of rows per sprite.
- `#define DEMOCOLS 18`
Defines the maximum number of columns per sprite.
- `#define DEMOFRAMETIME 67`
Defines for how many millisecs a sprite should stay on screen in the demo.

Functions

- void `usage` (char *programe)
- void `demo` (char *filename)
- int `main` (int argc, char **argv)

7.4.1 Macro Definition Documentation

7.4.1.1 DEMOCOLS

```
#define DEMOCOLS 18
```

Defines the maximum number of columns per sprite.

7.4.1.2 DEMOFRAMES

```
#define DEMOFRAMES 30
```

Defines the number of sprites in the demo.

7.4.1.3 DEMOFRAMETIME

```
#define DEMOFRAMETIME 67
```

Defines for how many millisecs a sprite should stay on screen in the demo.

7.4.1.4 DEMOROWS

```
#define DEMOROWS 18
```

Defines the maximum number of rows per sprite.

7.4.2 Function Documentation

7.4.2.1 demo()

```
void demo (
    char * filename )
```

7.4.2.2 main()

```
int main (
    int argc,
    char ** argv )
```

7.4.2.3 usage()

```
void usage (
    char * progname )
```

7.5 sprites4curses/documentation/README.md File Reference

7.6 sprites4curses/README.md File Reference

7.7 sprites4curses/palette-README.md File Reference

7.8 sprites4curses/png_resize.py File Reference

Program that resizes pngs to a desired size and overwrites them.

Namespaces

- namespace [png_resize](#)
Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

Functions

- def [png_resize.usage](#) ()
Prints correct invocation.
- def [png_resize.resize_sprites](#) (directory, targetSizeX, targetSizeY)
Resizes all png files in the passed directory to the specified size.
- def [png_resize.main](#) (argv)
Main program entry.

7.8.1 Detailed Description

Program that resizes pngs to a desired size and overwrites them.

7.8.2 Description

The program overwrites the passed pngs with the resized version.

Program expects the spritesheet filename as first argument, the sprite width as second arg, the sprite height as third arg.

7.8.3 Libraries/Modules

- Pillow (<https://pillow.readthedocs.io/en/stable/>)
 - Access to image manipulation functions.
- sys standard library (<https://docs.python.org/3/library/sys.html>)
 - Access to command line arguments.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.

7.8.4 Notes

- The pngs are overwritten by default.

7.8.5 TODO

- Offer option to output to new files and not overwrite.

7.8.6 Author(s)

- Created by jgabaut on 24/02/2022.
- Modified by jgabaut on 24/02/2022.

7.9 sprites4curses/sheet_converter.py File Reference

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

Namespaces

- namespace `sheet_converter`
Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

Functions

- def `sheet_converter.usage` ()
Prints correct invocation.
- def `sheet_converter.convert_spritesheet` (filename, spriteSizeX, spriteSizeY, separatorSize, startCoords)
Converts a spritesheet to a 3D char array representation of pixel color and then prints it with the needed brackets and commas.
- def `sheet_converter.main` (argv)
Main program entry.

7.9.1 Detailed Description

Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel.

7.9.2 Description

The program supports 8 colors at the moment. The png parsing uses Pillow, and the mapping is done against a preset color list. The list is described in palette.gpl to aid in exporting images with the correct color indexing.

Program expects the spritesheet filename as first argument, the sprite width as second arg, the sprite height as third, separator size (thickness) as fourth, a 0 or 1 for starting coords of the first sprite (0 if sheet has no edge separator) as fifth argument.

7.9.3 Libraries/Modules

- Pillow (<https://pillow.readthedocs.io/en/stable/>)
 - Access to image manipulation functions.
- sys standard library (<https://docs.python.org/3/library/sys.html>)
 - Access to command line arguments.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.

7.9.4 Notes

- Color map should have the same order as the palette used to index the sprites.

7.9.5 TODO

- The limitation to 8 colors will be overcome soon.

7.9.6 Author(s)

- Created by jgabaut on 24/02/2022.
- Modified by jgabaut on 24/02/2022.

7.10 sprites4curses/sprites.py File Reference

Program that parses pngs from a passed directory, to encode their color to a char per pixel.

Namespaces

- namespace `sprites`
Program that parses pngs from a passed directory, to encode their color to a char per pixel.

Functions

- def `sprites.usage` ()
Prints correct invocation.
- def `sprites.color_distance` (c1, c2)
Calculates the distance in color between two rgb tuples.
- def `sprites.create_palette` (colors)
Creates a palette by adding the colors passed as an array, then pads the palette.
- def `sprites.convert_sprite` (file)
Takes a image file and converts each pixel to a char representation of its color (closest match to CHAR_MAP).
- def `sprites.print_converted_sprites` (direc)
Takes a directory containing image file and calls convert_sprite on each one.
- def `sprites.main` (argv)
Main program entry.

7.10.1 Detailed Description

Program that parses pngs from a passed directory, to encode their color to a char per pixel.

7.10.2 Description

The program supports 8 colors at the moment. The png parsing uses Pillow, and the mapping is done against a preset color list. The list is described in palette.gpl to aid in exporting images with the correct color indexing.

7.10.3 Libraries/Moodules

- Pillow (<https://pillow.readthedocs.io/en/stable/>)
 - Access to image manipulation functions.
- sys standard library (<https://docs.python.org/3/library/sys.html>)
 - Access to command line arguments.
- glob standard library (<https://docs.python.org/3/library/glob.html>)
 - Access to pattern expansion.
- re standard library (<https://docs.python.org/3/library/re.html>)
 - Access to regular expressions.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.

7.10.4 Notes

- Color map should have the same order as the palette used to index the sprites.

7.10.5 TODO

- The limitation to 8 colors will be overcome soon.

7.10.6 Author(s)

- Created by jgabaut on 24/02/2022.
- Modified by jgabaut on 24/02/2022.

Index

- animate.c
 - animate_file, [17](#)
 - init_s4c_color_pairs, [17](#)
 - load_sprites, [18](#)
- animate.h
 - animate_file, [20](#)
 - BLACK, [18](#)
 - BLUE, [19](#)
 - CYAN, [19](#)
 - GREEN, [19](#)
 - init_s4c_color_pairs, [20](#)
 - load_sprites, [20](#)
 - MAGENTA, [19](#)
 - MAXCOLS, [19](#)
 - MAXFRAMES, [19](#)
 - MAXROWS, [19](#)
 - RED, [20](#)
 - WHITE, [20](#)
 - YELLOW, [20](#)
- animate_file
 - animate.c, [17](#)
 - animate.h, [20](#)
- BLACK
 - animate.h, [18](#)
- BLUE
 - animate.h, [19](#)
- color_distance
 - sprites, [14](#)
- convert_sprite
 - sprites, [14](#)
- convert_spritesheet
 - sheet_converter, [12](#)
- create_palette
 - sprites, [14](#)
- CYAN
 - animate.h, [19](#)
- demo
 - demo.c, [22](#)
- demo.c
 - demo, [22](#)
 - DEMOCOLS, [22](#)
 - DEMOFRAMES, [22](#)
 - DEMOFRAMETIME, [22](#)
 - DEMOROWS, [22](#)
 - main, [22](#)
 - usage, [23](#)
- DEMOCOLS
 - demo.c, [22](#)
- DEMOFRAMES
 - demo.c, [22](#)
- DEMOFRAMETIME
 - demo.c, [22](#)
- DEMOROWS
 - demo.c, [22](#)
- GREEN
 - animate.h, [19](#)
- init_s4c_color_pairs
 - animate.c, [17](#)
 - animate.h, [20](#)
- load_sprites
 - animate.c, [18](#)
 - animate.h, [20](#)
- MAGENTA
 - animate.h, [19](#)
- main
 - demo.c, [22](#)
 - png_resize, [11](#)
 - sheet_converter, [13](#)
 - sprites, [15](#)
- MAXCOLS
 - animate.h, [19](#)
- MAXFRAMES
 - animate.h, [19](#)
- MAXROWS
 - animate.h, [19](#)
- png_resize, [11](#)
 - main, [11](#)
 - resize_sprites, [11](#)
 - usage, [12](#)
- print_converted_sprites
 - sprites, [15](#)
- RED
 - animate.h, [20](#)
- resize_sprites
 - png_resize, [11](#)
- sheet_converter, [12](#)
 - convert_spritesheet, [12](#)
 - main, [13](#)
 - usage, [13](#)
- sprites, [13](#)
 - color_distance, [14](#)

- [convert_sprite](#), [14](#)
 - [create_palette](#), [14](#)
 - [main](#), [15](#)
 - [print_converted_sprites](#), [15](#)
 - [usage](#), [15](#)
- [sprites4curses/animate.c](#), [17](#)
- [sprites4curses/animate.h](#), [18](#), [21](#)
- [sprites4curses/demo.c](#), [21](#)
- [sprites4curses/documentation/README.md](#), [23](#)
- [sprites4curses/palette-README.md](#), [23](#)
- [sprites4curses/png_resize.py](#), [23](#)
- [sprites4curses/README.md](#), [23](#)
- [sprites4curses/sheet_converter.py](#), [24](#)
- [sprites4curses/sprites.py](#), [26](#)
- [usage](#)
 - [demo.c](#), [23](#)
 - [png_resize](#), [12](#)
 - [sheet_converter](#), [13](#)
 - [sprites](#), [15](#)
- WHITE
 - [animate.h](#), [20](#)
- YELLOW
 - [animate.h](#), [20](#)