

sprites4curses

0.2.2

Generated by Doxygen 1.9.7

1 Contributor Covenant Code of Conduct	1
1.1 Our Pledge	1
1.2 Our Standards	1
1.3 Enforcement Responsibilities	2
1.4 Scope	2
1.5 Enforcement	2
1.6 Enforcement Guidelines	2
1.6.1 1. Correction	2
1.6.2 2. Warning	2
1.6.3 3. Temporary Ban	3
1.6.4 4. Permanent Ban	3
1.7 Attribution	3
2 palette.gpl	5
3 sprites4curses	7
3.1 sprites.py	7
3.1.1 sheet_converter.py	7
3.1.2 cut_sheet.py	7
3.1.3 png_resize.py	7
3.2 palette.gpl	8
3.3 animate.c and animate.h	8
3.3.1 demo.c	8
3.4 Usage	8
3.4.0.1 This overwrites the source pngs, so be careful.	9
3.4.1 The demo program is meant to show how to correctly call <code>animate_file()</code> from <code>animate.h</code> .	9
4 Namespace Index	11
4.1 Namespace List	11
5 File Index	13
5.1 File List	13
6 Namespace Documentation	15
6.1 cut_sheet Namespace Reference	15
6.1.1 Detailed Description	15
6.1.2 Function Documentation	15
6.1.2.1 <code>color_distance()</code>	15
6.1.2.2 <code>cut_spritesheet()</code>	16
6.2 png_resize Namespace Reference	16
6.2.1 Detailed Description	17
6.2.2 Function Documentation	17
6.2.2.1 <code>resize_sprites()</code>	17
6.3 sheet_converter Namespace Reference	17

6.3.1 Detailed Description	18
6.3.2 Function Documentation	18
6.3.2.1 color_distance()	18
6.3.2.2 convert_spritesheet()	18
6.4 sprites Namespace Reference	19
6.4.1 Detailed Description	19
6.4.2 Function Documentation	19
6.4.2.1 color_distance()	19
6.4.2.2 convert_sprite()	20
6.4.2.3 print_converted_sprites()	20
7 File Documentation	21
7.1 animate.h	21
7.2 sprites4curses/cut_sheet.py File Reference	21
7.2.1 Detailed Description	22
7.2.2 Description	22
7.2.3 Libraries/Moodules	22
7.2.4 Notes	22
7.2.5 TODO	23
7.2.6 Author(s)	23
7.3 sprites4curses/png_resize.py File Reference	23
7.3.1 Detailed Description	23
7.3.2 Description	23
7.3.3 Libraries/Moodules	23
7.3.4 Notes	24
7.3.5 TODO	24
7.3.6 Author(s)	24
7.4 sprites4curses/sheet_converter.py File Reference	24
7.4.1 Detailed Description	24
7.4.2 Description	25
7.4.3 Libraries/Moodules	25
7.4.4 Notes	25
7.4.5 TODO	25
7.4.6 Author(s)	25
7.5 sprites4curses/sprites.py File Reference	25
7.5.1 Detailed Description	26
7.5.2 Description	26
7.5.3 Libraries/Moodules	26
7.5.4 Notes	27
7.5.5 TODO	27
7.5.6 Author(s)	27
Index	29

Chapter 1

Contributor Covenant Code of Conduct

1.1 Our Pledge

We as members, contributors, and leaders pledge to make participation in our community a harassment-free experience for everyone, regardless of age, body size, visible or invisible disability, ethnicity, sex characteristics, gender identity and expression, level of experience, education, socio-economic status, nationality, personal appearance, race, religion, or sexual identity and orientation.

We pledge to act and interact in ways that contribute to an open, welcoming, diverse, inclusive, and healthy community.

1.2 Our Standards

Examples of behavior that contributes to a positive environment for our community include:

- Demonstrating empathy and kindness toward other people
- Being respectful of differing opinions, viewpoints, and experiences
- Giving and gracefully accepting constructive feedback
- Accepting responsibility and apologizing to those affected by our mistakes, and learning from the experience
- Focusing on what is best not just for us as individuals, but for the overall community

Examples of unacceptable behavior include:

- The use of sexualized language or imagery, and sexual attention or advances of any kind
- Trolling, insulting or derogatory comments, and personal or political attacks
- Public or private harassment
- Publishing others' private information, such as a physical or email address, without their explicit permission
- Other conduct which could reasonably be considered inappropriate in a professional setting

1.3 Enforcement Responsibilities

Community leaders are responsible for clarifying and enforcing our standards of acceptable behavior and will take appropriate and fair corrective action in response to any behavior that they deem inappropriate, threatening, offensive, or harmful.

Community leaders have the right and responsibility to remove, edit, or reject comments, commits, code, wiki edits, issues, and other contributions that are not aligned to this Code of Conduct, and will communicate reasons for moderation decisions when appropriate.

1.4 Scope

This Code of Conduct applies within all community spaces, and also applies when an individual is officially representing the community in public spaces. Examples of representing our community include using an official e-mail address, posting via an official social media account, or acting as an appointed representative at an online or offline event.

1.5 Enforcement

Instances of abusive, harassing, or otherwise unacceptable behavior may be reported to the community leaders responsible for enforcement at All complaints will be reviewed and investigated promptly and fairly.

All community leaders are obligated to respect the privacy and security of the reporter of any incident.

1.6 Enforcement Guidelines

Community leaders will follow these Community Impact Guidelines in determining the consequences for any action they deem in violation of this Code of Conduct:

1.6.1 1. Correction

Community Impact: Use of inappropriate language or other behavior deemed unprofessional or unwelcome in the community.

Consequence: A private, written warning from community leaders, providing clarity around the nature of the violation and an explanation of why the behavior was inappropriate. A public apology may be requested.

1.6.2 2. Warning

Community Impact: A violation through a single incident or series of actions.

Consequence: A warning with consequences for continued behavior. No interaction with the people involved, including unsolicited interaction with those enforcing the Code of Conduct, for a specified period of time. This includes avoiding interactions in community spaces as well as external channels like social media. Violating these terms may lead to a temporary or permanent ban.

1.6.3 3. Temporary Ban

Community Impact: A serious violation of community standards, including sustained inappropriate behavior.

Consequence: A temporary ban from any sort of interaction or public communication with the community for a specified period of time. No public or private interaction with the people involved, including unsolicited interaction with those enforcing the Code of Conduct, is allowed during this period. Violating these terms may lead to a permanent ban.

1.6.4 4. Permanent Ban

Community Impact: Demonstrating a pattern of violation of community standards, including sustained inappropriate behavior, harassment of an individual, or aggression toward or disparagement of classes of individuals.

Consequence: A permanent ban from any sort of public interaction within the community.

1.7 Attribution

This Code of Conduct is adapted from the [Contributor Covenant](https://www.contributor-covenant.org/version/2/0/code_of_conduct.html), version 2.0, available at https://www.contributor-covenant.org/version/2/0/code_of_conduct.html.

Community Impact Guidelines were inspired by [Mozilla's code of conduct enforcement ladder](#).

For answers to common questions about this code of conduct, see the FAQ at <https://www.contributor-covenant.org/faq>. Translations are available at <https://www.contributor-covenant.org/translations>.

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.

3.1.1 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.

3.1.2 cut_sheet.py

This is a python script that cuts a single PNG spritesheet to a number of sprites, and puts them in the passed directory.

It expects as arguments the spritesheet file name, the output directory 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.

3.1.3 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.

3.2 palette.gpl

This is a GIMP palette file. It's used by the library to initialise the color pairs for curses to display the sprites. It's also useful in the first place for exporting PNG with the correct color alignment. Info on how to use it are in the `palette-Readme.md` file.

3.3 animate.c and animate.h

This is a C library offering some functions to display an animation read from a formatted text file.

`animate_sprites()` is useful in a initialised WINDOW, it boxes the window and displays the animation snugly.

`animate_sprites_at_coords()` does the same, but has 2 more parameters to start displaying at any coord in a window big enough to fit the animation.

You can look at the `demo.c` program to see how you can request the animation after setup. The file format expected is compatible with [sprites.py](#) specs.

Since it needs support from terminal capabilities, it may return some errors if your terminal doesn't offer the needed options.

At the moment your solution is to change terminal or help investigate your issues by forcing the check to pass, I may add an unsafe option to do this in a later version.

3.3.1 demo.c

This is a demo program showing how to use the animate library functions.

3.4 Usage

To use the python scripts you need to install Pillow:

3.4.0.0.1 `<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:

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

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

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

3.4.0.0.4 `<tt>python sheet_converter.py <sheet file> <sprite width> <sprite height> <separator thickness> <first sprite left corner X> <first sprite LC Y> > file.txt</tt>`

- To run the sheet cutter script, give the sheet png file and the output directory:

3.4.0.0.5 `<tt>python cut_sheet.py <sheet file> <output dir> <sprite width> <sprite height> <separator thickness> <first sprite left corner X> <first sprite LC Y> </tt>`

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

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

3.4.0.1 This overwrites the source pngs, so be careful.

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

- To run the C demo program, you do:

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

3.4.1.0.2 `autotoc_md31`

3.4.1.0.3 `<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:

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

3.4.1.0.5 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 documented namespaces with brief descriptions:

cut_sheet	Program that cut a passed png spritesheet, and saves the sprites to a passed directory	15
png_resize	Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel .	16
sheet_converter	Program that parses pngs from a passed spritesheet, to encode their color to a char per pixel .	17
sprites	Program that parses pngs from a passed directory, to encode their color to a char per pixel . .	19

Chapter 5

File Index

5.1 File List

Here is a list of all documented files with brief descriptions:

sprites4curses/ animate.h	21
sprites4curses/ cut_sheet.py	
Program that cut a passed png spritesheet, and saves the sprites to a passed directory	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 . .	25

Chapter 6

Namespace Documentation

6.1 cut_sheet Namespace Reference

Program that cut a passed png spritesheet, and saves the sprites to a passed directory.

Functions

- **usage** ()
Prints correct invocation.
- [color_distance](#) (c1, c2)
Calculates the distance in color between two rgb tuples.
- [cut_spritesheet](#) (filename, output_directory, spriteSizeX, spriteSizeY, separatorSize, startX, startY)
Converts a spritesheet to a set of individual sprite images.
- **main** (argv)
Main program entry.

6.1.1 Detailed Description

Program that cut a passed png spritesheet, and saves the sprites to a passed directory.

6.1.2 Function Documentation

6.1.2.1 color_distance()

```
cut_sheet.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.

6.1.2.2 cut_spritesheet()

```
cut_sheet.cut_spritesheet (
    filename,
    output_directory,
    spriteSizeX,
    spriteSizeY,
    separatorSize,
    startX,
    startY )
```

Converts a spritesheet to a set of individual sprite images.

Parameters

<i>filename</i>	The input spritesheet file.
<i>output_directory</i>	The directory where output images will be saved.
<i>spriteSizeX</i>	The sprite width.
<i>spriteSizeY</i>	The sprite height.
<i>separatorSize</i>	Thickess of separator pixels.
<i>startX</i>	X coord of left corner of first sprite.
<i>startY</i>	Y coord of left corner of first sprite.

6.2 png_resize Namespace Reference

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

Functions

- **usage** ()
Prints correct invocation.
- **resize_sprites** (directory, targetSizeX, targetSizeY)
Resizes all png files in the passed directory to the specified size.
- **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 `resize_sprites()`

```
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.

6.3 sheet_converter Namespace Reference

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

Functions

- **usage** ()
Prints correct invocation.
- **color_distance** (c1, c2)
Calculates the distance in color between two rgb tuples.
- **convert_spritesheet** (filename, spriteSizeX, spriteSizeY, separatorSize, startX, startY)
Converts a spritesheet to a 3D char array representation of pixel color and then prints it with the needed brackets and commas.
- **main** (argv)
Main program entry.

Variables

- str **FILE_VERSION** = "0.1.2"
The file format version.

6.3.1 Detailed Description

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

6.3.2 Function Documentation

6.3.2.1 `color_distance()`

```
sheet_converter.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_spritesheet()`

```
sheet_converter.convert_spritesheet (
    filename,
    spriteSizeX,
    spriteSizeY,
    separatorSize,
    startX,
    startY )
```

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>startX</i>	X coord of left corner of first sprite.
<i>startY</i>	Y coord of left corner of first sprite.

6.4 sprites Namespace Reference

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

Functions

- **usage** ()
Prints correct invocation.
- **color_distance** (c1, c2)
Calculates the distance in color between two rgb tuples.
- **convert_sprite** (file)
Takes a image file and converts each pixel to a char representation of its color (closest match to CHAR_MAP).
- **print_converted_sprites** (direc)
Takes a directory containing image file and calls convert_sprite on each one.
- **main** (argv)
Main program entry.

Variables

- str **FILE_VERSION** = "0.1.2"
The file format version.

6.4.1 Detailed Description

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

6.4.2 Function Documentation

6.4.2.1 color_distance()

```
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.

6.4.2.2 convert_sprite()

```
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.

6.4.2.3 print_converted_sprites()

```
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.
--------------	--

Chapter 7

File Documentation

7.1 animate.h

```
00001 #ifndef S4C_ANIMATE_H
00002 #define S4C_ANIMATE_H
00003
00004 #include <ncurses.h>
00005
00006 #define MAX_COLORS 256
00007 #define MAX_COLOR_NAME_LEN 256
00009 // These define constants for the colors prepared by init_s4c_color_pairs().
00010 #define S4C_BLACK 9
00011 #define S4C_RED 10
00012 #define S4C_BRIGHT_GREEN 11
00013 #define S4C_BRIGHT_YELLOW 12
00014 #define S4C_BLUE 13
00015 #define S4C_MAGENTA 14
00016 #define S4C_CYAN 15
00017 #define S4C_WHITE 16
00018 #define S4C_ORANGE 17
00019 #define S4C_LIGHT_BROWN 18
00020 #define S4C_DARK_BROWN 19
00021 #define S4C_PURPLE 20
00022 #define S4C_DARK_GREEN 21
00023 #define S4C_LIGHT_YELLOW 22
00024 #define S4C_LIGHT_BLUE 23
00025
00026 #define MAX_LINE_LENGTH 1024
00027
00028 #define MAXFRAMES 121
00029 #define MAXROWS 26
00030 #define MAXCOLS 84
00032 #define S4C_ERR_SMALL_WIN -1
00033 #define S4C_ERR_FILEVERSION -2
00034 #define S4C_ERR_LOADSPRITES -3
00035 #define S4C_ERR_TERMCOLOR -4
00036 #define S4C_ERR_TERMCHANGECOLOR -5
00037 #define S4C_ERR_CURSOR -6
00039 void init_s4c_color_pairs(FILE* palette_file);
00040 static void print_spriteline(WINDOW* win, char* line, int curr_line_num, int line_length, int startX);
00041 int load_sprites(char sprites[MAXFRAMES][MAXROWS][MAXCOLS], FILE* file, int rows, int columns);
00042 int animate_sprites(char sprites[MAXFRAMES][MAXROWS][MAXCOLS], WINDOW* w, int repetitions, int
    frametime, int num_frames, int frameheight, int framewidth);
00043 int animate_sprites_at_coords(char sprites[MAXFRAMES][MAXROWS][MAXCOLS], WINDOW* w, int repetitions,
    int frametime, int num_frames, int frameheight, int framewidth, int startX, int startY);
00044
00045 #endif
```

7.2 sprites4curses/cut_sheet.py File Reference

Program that cut a passed png spritesheet, and saves the sprites to a passed directory.

Namespaces

- namespace [cut_sheet](#)

Program that cut a passed png spritesheet, and saves the sprites to a passed directory.

Functions

- **cut_sheet.usage** ()
Prints correct invocation.
- **cut_sheet.color_distance** (c1, c2)
Calculates the distance in color between two rgb tuples.
- **cut_sheet.cut_spritesheet** (filename, output_directory, spriteSizeX, spriteSizeY, separatorSize, startX, startY)
Converts a spritesheet to a set of individual sprite images.
- **cut_sheet.main** (argv)
Main program entry.

7.2.1 Detailed Description

Program that cut a passed png spritesheet, and saves the sprites to a passed directory.

7.2.2 Description

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, then the output directory, the sprite width, the sprite height, separator size (thickness), X coord of left corner of the first sprite (0 if sheet has no edge separator), Y coord of left corner of the first sprite (0 if sheet has no edge separator).

7.2.3 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.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.
- math standard library (<https://docs.python.org/3/library/math.html>)
 - Access to sqrt.

7.2.4 Notes

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

7.2.5 TODO

- Check if the encoded value is exceeding latin literals.

7.2.6 Author(s)

- Created by jgabaut on 04/03/2023.
- Modified by jgabaut on 04/03/2023.

7.3 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

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

7.3.1 Detailed Description

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

7.3.2 Description

The program overwrites the passed pngs with the resized version.

Program expects the spritesheet filename as first argument, then the sprite width, the sprite heigth.

7.3.3 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.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.

7.3.4 Notes

- The pngs are overwritten by default.

7.3.5 TODO

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

7.3.6 Author(s)

- Created by jgabaut on 24/02/2023.
- Modified by jgabaut on 04/03/2023.

7.4 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

- **sheet_converter.usage** ()
Prints correct invocation.
- [sheet_converter.color_distance](#) (c1, c2)
Calculates the distance in color between two rgb tuples.
- [sheet_converter.convert_spritesheet](#) (filename, spriteSizeX, spriteSizeY, separatorSize, startX, startY)
Converts a spritesheet to a 3D char array representation of pixel color and then prints it with the needed brackets and commas.
- **sheet_converter.main** (argv)
Main program entry.

Variables

- str **sheet_converter.FILE_VERSION** = "0.1.2"
The file format version.

7.4.1 Detailed Description

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

7.4.2 Description

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, then the sprite width, the sprite height, separator size (thickness), X coord of left corner of the first sprite (0 if sheet has no edge separator), Y coord of left corner of the first sprite (0 if sheet has no edge separator).

7.4.3 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.
- os standard library (<https://docs.python.org/3/library/os.html>)
 - Access to program name.
- math standard library (<https://docs.python.org/3/library/math.html>)
 - Access to sqrt.

7.4.4 Notes

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

7.4.5 TODO

- Check if the encoded value is exceeding latin literals.

7.4.6 Author(s)

- Created by jgabaut on 24/02/2023.
- Modified by jgabaut on 04/03/2023.

7.5 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

- **sprites.usage** ()
Prints correct invocation.
- **sprites.color_distance** (c1, c2)
Calculates the distance in color between two rgb tuples.
- **sprites.convert_sprite** (file)
Takes a image file and converts each pixel to a char representation of its color (closest match to CHAR_MAP).
- **sprites.print_converted_sprites** (direc)
Takes a directory containing image file and calls convert_sprite on each one.
- **sprites.main** (argv)
Main program entry.

Variables

- str **sprites.FILE_VERSION** = "0.1.2"
The file format version.

7.5.1 Detailed Description

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

7.5.2 Description

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.

7.5.3 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.
- math standard library (<https://docs.python.org/3/library/math.html>)
 - Access to sqrt.

7.5.4 Notes

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

7.5.5 TODO

- Check if the encoded value is exceeding latin literals.

7.5.6 Author(s)

- Created by jgabaut on 24/02/2023.
- Modified by jgabaut on 04/03/2023.

Index

- color_distance
 - cut_sheet, [15](#)
 - sheet_converter, [18](#)
 - sprites, [19](#)
- convert_sprite
 - sprites, [20](#)
- convert_spritesheet
 - sheet_converter, [18](#)
- cut_sheet, [15](#)
 - color_distance, [15](#)
 - cut_spritesheet, [16](#)
- cut_spritesheet
 - cut_sheet, [16](#)
- png_resize, [16](#)
 - resize_sprites, [17](#)
- print_converted_sprites
 - sprites, [20](#)
- resize_sprites
 - png_resize, [17](#)
- sheet_converter, [17](#)
 - color_distance, [18](#)
 - convert_spritesheet, [18](#)
- sprites, [19](#)
 - color_distance, [19](#)
 - convert_sprite, [20](#)
 - print_converted_sprites, [20](#)
- sprites4curses/animate.h, [21](#)
- sprites4curses/cut_sheet.py, [21](#)
- sprites4curses/png_resize.py, [23](#)
- sprites4curses/sheet_converter.py, [24](#)
- sprites4curses/sprites.py, [25](#)