

Conway

Generated by Doxygen 1.9.1

1 File Index	1
1.1 File List	1
2 File Documentation	3
2.1 conway.c File Reference	3
2.1.1 Detailed Description	3
2.1.2 Function Documentation	3
2.1.2.1 advance_generation()	3
2.1.2.2 initialize_world()	4
2.1.2.3 print_world()	4
2.1.2.4 read_generations()	4
2.2 conway.h File Reference	5
2.2.1 Detailed Description	5
2.2.2 Macro Definition Documentation	5
2.2.2.1 ALIVE	5
2.2.2.2 ALIVE_CHAR	6
2.2.2.3 DEAD	6
2.2.2.4 DEAD_CHAR	6
2.2.2.5 OUTSIDE_STATE	6
2.2.2.6 SEPARATOR	6
2.2.2.7 WORLD_HEIGHT	6
2.2.2.8 WORLD_WIDTH	6
2.2.3 Function Documentation	6
2.2.3.1 advance_generation()	6
2.2.3.2 initialize_world()	7
2.2.3.3 print_world()	7
2.2.3.4 read_generations()	7
2.3 helper_functions.c File Reference	7
2.3.1 Detailed Description	7
2.3.2 Function Documentation	8
2.3.2.1 clear_screen()	8
2.3.2.2 flush_stdin()	8
2.4 helper_functions.h File Reference	8
2.4.1 Detailed Description	8
2.4.2 Function Documentation	8
2.4.2.1 clear_screen()	8
2.4.2.2 flush_stdin()	9
2.5 main.c File Reference	9
2.5.1 Detailed Description	9
2.5.2 Function Documentation	9
2.5.2.1 main()	9
Index	11

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

conway.c	Implements Conway's game of life	3
conway.h	Implements Conway's game of life	5
helper_functions.c	Group 7 helper functions	7
helper_functions.h	Group 7 helper functions	8
main.c	Plays Conway's Game of Life	9

Chapter 2

File Documentation

2.1 conway.c File Reference

Implements Conway's game of life.

```
#include "conway.h"
#include "helper_functions.h"
#include <stdio.h>
```

Functions

- void [initialize_world](#) (int world[][[WORLD_WIDTH](#)])
Handles world initialization. Allows the user to select different input methods to customize the world to load. Includes instructions on how to manually load a world file.
- void [advance_generation](#) (int world_to_advance[][[WORLD_WIDTH](#)])
Advances the current generation, analyzing the future state of all cells.
- int [read_generations](#) (void)
Reads an unsigned number from stdin.
- void [print_world](#) (int world[][[WORLD_WIDTH](#)])
Prints the world state to stdout in a polished format.

2.1.1 Detailed Description

Implements Conway's game of life.

Author

Grupo 7

2.1.2 Function Documentation

2.1.2.1 [advance_generation\(\)](#)

```
void advance_generation (
    int world_to_advance[ ][WORLD\_WIDTH] )
```

Advances the current generation, analyzing the future state of all cells.

Parameters

<i>world_to_advance</i>	world matrix where the current state is and the future state will be saved before returning.
-------------------------	--

Note

This function implements John Conway's Game of Life rules

- A living cell with exactly two or three living neighbors stays alive
- A dead cell with exactly three living neighbors becomes alive

2.1.2.2 initialize_world()

```
void initialize_world (
    int world[][WORLD_WIDTH] )
```

Handles world initialization. Allows the user to select different input methods to customize the world to load. Includes instructions on how to manually load a world file.

Parameters

<i>world</i>	matrix where the initial world state is saved.
--------------	--

2.1.2.3 print_world()

```
void print_world (
    int world[][WORLD_WIDTH] )
```

Prints the world state to stdout in a polished format.

Parameters

<i>world</i>	world to print to stdout
--------------	--------------------------

2.1.2.4 read_generations()

```
int read_generations (
    void )
```

Reads an unsigned number from stdin.

Returns

1 if the first char is '\n', -1 if it's 'q', the input number if it's valid, 0 otherwise.

2.2 conway.h File Reference

Implements Conway's game of life.

Macros

- `#define WORLD_WIDTH 25`
Ancho del mundo. Número de columnas.
- `#define WORLD_HEIGHT 15`
Altura del mundo. Número de filas.
- `#define OUTSIDE_STATE DEAD`
Estado de las celdas del borde externo.
- `#define ALIVE_CHAR '*'`
Caracter que representa una celda viva.
- `#define DEAD_CHAR ' '`
Caracter que representa una celda muerta.
- `#define SEPARATOR '|'`
Separador de columnas.
- `#define ALIVE 1`
Valor interno de una celda viva.
- `#define DEAD 0`
Valor interno de una celda muerta.

Functions

- `int read_generations (void)`
Reads an unsigned number from stdin.
- `void advance_generation (int world_to_advance[][WORLD_WIDTH])`
Advances the current generation, analyzing the future state of all cells.
- `void print_world (int world[][WORLD_WIDTH])`
Prints the world state to stdout in a polished format.
- `void initialize_world (int world[][WORLD_WIDTH])`
Handles world initialization. Allows the user to select different input methods to customize the world to load. Includes instructions on how to manually load a world file.

2.2.1 Detailed Description

Implements Conway's game of life.

Author

Grupo 7

2.2.2 Macro Definition Documentation

2.2.2.1 ALIVE

```
#define ALIVE 1
```

Valor interno de una celda viva.

2.2.2.2 ALIVE_CHAR

```
#define ALIVE_CHAR '*'
```

Caracter que representa una celda viva.

2.2.2.3 DEAD

```
#define DEAD 0
```

Valor interno de una celda muerta.

2.2.2.4 DEAD_CHAR

```
#define DEAD_CHAR ' '
```

Caracter que representa una celda muerta.

2.2.2.5 OUTSIDE_STATE

```
#define OUTSIDE_STATE DEAD
```

Estado de las celdas del borde externo.

2.2.2.6 SEPARATOR

```
#define SEPARATOR '|'
```

Separador de columnas.

2.2.2.7 WORLD_HEIGHT

```
#define WORLD_HEIGHT 15
```

Altura del mundo. Número de filas.

2.2.2.8 WORLD_WIDTH

```
#define WORLD_WIDTH 25
```

Ancho del mundo. Número de columnas.

2.2.3 Function Documentation

2.2.3.1 advance_generation()

```
void advance_generation (
    int world_to_advance[][WORLD_WIDTH] )
```

Advances the current generation, analyzing the future state of all cells.

Parameters

<i>world_to_advance</i>	world matrix where the current state is and the future state will be saved before returning.
-------------------------	--

Note

This function implements John Conway's Game of Life rules

- A living cell with exactly two or three living neighbors stays alive

- A dead cell with exactly three living neighbors becomes alive

2.2.3.2 initialize_world()

```
void initialize_world (
    int world[][WORLD_WIDTH] )
```

Handles world initialization. Allows the user to select different input methods to customize the world to load. Includes instructions on how to manually load a world file.

Parameters

<i>world</i>	matrix where the initial world state is saved.
--------------	--

2.2.3.3 print_world()

```
void print_world (
    int world[][WORLD_WIDTH] )
```

Prints the world state to stdout in a polished format.

Parameters

<i>world</i>	world to print to stdout
--------------	--------------------------

2.2.3.4 read_generations()

```
int read_generations (
    void )
```

Reads an unsigned number from stdin.

Returns

1 if the first char is '\n', -1 if it's 'q', the input number if it's valid, 0 otherwise.

2.3 helper_functions.c File Reference

Group 7 helper functions.

```
#include <stdio.h>
#include "helper_functions.h"
```

Functions

- void [clear_screen](#) (void)
Clears terminal in an OS-specific basis.
- void [flush_stdin](#) (char limit)
Flushes the stdin buffer. Takes characters from stdin until limit is found.

2.3.1 Detailed Description

Group 7 helper functions.

Author

Grupo 7

2.3.2 Function Documentation

2.3.2.1 clear_screen()

```
void clear_screen (
    void )
```

Clears terminal in an OS-specific basis.

2.3.2.2 flush_stdin()

```
void flush_stdin (
    char limit )
```

Flushes the stdin buffer. Takes characters from stdin until limit is found.

Parameters

<i>limit</i>	Character that indicates where to stop flushing
--------------	---

2.4 helper_functions.h File Reference

Group 7 helper functions.

```
#include <stdlib.h>
```

Functions

- void [clear_screen](#) (void)
Clears terminal in an OS-specific basis.
- void [flush_stdin](#) (char limit)
Flushes the stdin buffer. Takes characters from stdin until limit is found.

2.4.1 Detailed Description

Group 7 helper functions.

Author

Grupo 7

2.4.2 Function Documentation

2.4.2.1 clear_screen()

```
void clear_screen (
    void )
```

Clears terminal in an OS-specific basis.

2.4.2.2 flush_stdin()

```
void flush_stdin (
    char limit )
```

Flushes the stdin buffer. Takes characters from stdin until limit is found.

Parameters

<i>limit</i>	Character that indicates where to stop flushing
--------------	---

2.5 main.c File Reference

Plays Conway's Game of Life.

```
#include <stdio.h>
#include "helper_functions.h"
#include "conway.h"
```

Functions

- int [main](#) (void)
Main function.

2.5.1 Detailed Description

Plays Conway's Game of Life.

Author

Grupo 7

2.5.2 Function Documentation

2.5.2.1 main()

```
int main (
    void )
```

Main function.

Index

- advance_generation
 - conway.c, [3](#)
 - conway.h, [6](#)
- ALIVE
 - conway.h, [5](#)
- ALIVE_CHAR
 - conway.h, [5](#)
- clear_screen
 - helper_functions.c, [8](#)
 - helper_functions.h, [8](#)
- conway.c, [3](#)
 - advance_generation, [3](#)
 - initialize_world, [4](#)
 - print_world, [4](#)
 - read_generations, [4](#)
- conway.h, [5](#)
 - advance_generation, [6](#)
 - ALIVE, [5](#)
 - ALIVE_CHAR, [5](#)
 - DEAD, [6](#)
 - DEAD_CHAR, [6](#)
 - initialize_world, [7](#)
 - OUTSIDE_STATE, [6](#)
 - print_world, [7](#)
 - read_generations, [7](#)
 - SEPARATOR, [6](#)
 - WORLD_HEIGHT, [6](#)
 - WORLD_WIDTH, [6](#)
- DEAD
 - conway.h, [6](#)
- DEAD_CHAR
 - conway.h, [6](#)
- flush_stdin
 - helper_functions.c, [8](#)
 - helper_functions.h, [8](#)
- helper_functions.c, [7](#)
 - clear_screen, [8](#)
 - flush_stdin, [8](#)
- helper_functions.h, [8](#)
 - clear_screen, [8](#)
 - flush_stdin, [8](#)
- initialize_world
 - conway.c, [4](#)
 - conway.h, [7](#)
- main
 - main.c, [9](#)
 - main.c, [9](#)
 - main, [9](#)
- OUTSIDE_STATE
 - conway.h, [6](#)
- print_world
 - conway.c, [4](#)
 - conway.h, [7](#)
- read_generations
 - conway.c, [4](#)
 - conway.h, [7](#)
- SEPARATOR
 - conway.h, [6](#)
- WORLD_HEIGHT
 - conway.h, [6](#)
- WORLD_WIDTH
 - conway.h, [6](#)