

# Contents

<b>A</b>	<b>Introduction</b>	<b>2</b>
<b>B</b>	<b>Man Pages</b>	<b>2</b>
B.1	Accessing Man pages . . . . .	2
B.2	TRGadmin . . . . .	2
B.3	TRGserver . . . . .	3
B.4	TRGtesting . . . . .	3
B.5	Dependencies . . . . .	3
B.6	References . . . . .	4

# Administrative Manual

April 25, 2024

## A Introduction

Toadally Real Graphs is a multi-functional tool designed for graph visualization, bibliography searching, and graph recognition. Utilizing adjacency matrices, users can input graph data and visualize relationships between nodes. Additionally, our program includes administrative parsing capabilities for managing administrative data. Our program offers graph recognition functionality, allowing users to determine whether inputted graphs are recognizable or not.

## B Man Pages

### B.1 Accessing Man pages

There are three different man pages offering information on how to navigate the admin tool (TRGadmin), how to launch the server (TRGserver), as well as how to run tests (TRGtesting). All of these man pages can be accessed from the Documentation folder (NathanInc/DOCUMENTATION) by running the following command in the command line:

```
man ./[man page name]
Example: man ./TRGadmin
```

### B.2 TRGadmin

TRGadmin The admin tool is designed to allow administrators to quickly and easily make changes to the website. The tool will automatically determine which table the data is intended for based on the syntax.

Bibliography entries must be in a file exclusively for bibtex; however, you can add to multiple tables using a single file (again excluding the bibliography table) as long as the syntax for that entry matches the specified table.

For the bibliography entries must be added in the file as bibtex citations where each element is separated by a new line. This table can not contain additional "" besides the opening and closing ones! There must also be exactly one blank line between each citation in the file. For the other three tables, each new entry must be on a single line. Again, that entry's syntax must match the specified table.

When removing rows, any combination of tables can be modified from a single file. Each row to be removed must be on a single line, and all tables use the same syntax.

When updating cells, any combination of tables can be modified from a single file. Each cell change must be on a single line, and all tables use the same syntax.

Add file syntax for each table:

```
CHECK1$BIBKEY$FORMULA CHECK2$BIBKEY$FORMULA$FUNCTION SANITIZE$FORMULA
```

```
(bibtex example)
@article{BIBKEY,
  author= "author",
  title= "title",
  year="year"
}
```

RULE	SUBSTITUTIONS
Not	!
And	&
If Complete	C
If Regular	R
Vertex Connectivity	k
Edge Connectivity	l
Minimum Degree	d
Maximum Degree	D

Table 1: Substitutions

2. Remove files syntax for each table: TABLE\$BIBKEY
3. Update files syntax for each table: TABLE\$NEW\_VALUE\$COLUMN\$BIBKEY
4. For updating from table: SANTIZEN\$NEW\$COLUMN\$OLD

TABLE\$NEW\_VALUE\$COLUMN\$BIBKEY

OPTIONS -a, -add Add information to the database.

- d, -delete Delete a row from the specified table.
- f, -file Specify a file to import data from.
- h, -help Displays help menu.
- p, -print Print elements from the specified table.
- t, -table Select a specific table in the database
- u, -update Update a cell in the database.

### B.3 TRGserver

TRGserver For the formulas in our database, we defined our own syntax for adding condition statements. Formulas can only contain one conditional per formula, so boolean logic must be used if more are needed. Eval() evaluates python syntax, so in most cases all other syntax in the formulas can be used as such. However, We had to require certain python3 key words to be substituted to avoid conflicts with variable names. The statements before and after the colon must have a statement even if it is just True. The constructors used to draw the graphs are stored in builder.py. To add a constructor for a theorem, simply add a function that returns an adjacency matrix to the builder.py file and update the CHECKTWO table entry to reflect that function name.

To Run: ./TRGserver.py

### B.4 TRGtesting

TRGtesting This file provides test cases for the Api an database functions and the UI. 3 We used Selenium for the UI and pytest for everything else.

There are various ways to run the test cases.

For frontend testing: -make ui

For backend testing: -make test

For testing both simultaneously: -make

### B.5 Dependencies

Updated versions of the following libraries and applications are necessary to properly use all Toadally Real Graphs administrative tools...

- Argparse
- Base64
- Flask

- Jinja
- Math
- NetworkX
- Os
- PyGraphviz
- Pytest
- Python3
- Random
- Sys
- Uuid

## **B.6 References**

<https://www.overleaf.com>