

Recall Manual

July 22, 2020

Contents

1	Installation	2
1.1	Graphical Installation – Simplest	2
1.1.1	Downloading Installer	2
1.1.2	Unzipping Installer	2
1.1.3	Running Install Script	3
1.1.4	Potential Problems	4
1.2	Command-Line Installation – Most Reliable	4
1.2.1	Downloading Installer	4
1.2.2	Unzipping Installer	5
1.2.3	Running Install Script	6
1.2.4	potential problems	6
1.3	Manual Install – Not Recommended	7
1.3.1	Why Manually Install?	7
1.3.2	Downloading Necessary Files	7
1.3.3	Storing Executable	7
1.3.4	Adding Recall to \$PATH	7
1.3.5	Creating Launchers	7
1.4	Why the Install Script isn't Working	8
1.4.1	Recall Will not Run when Invoked from the Command-Line	8
1.4.2	Recall was not added to the Applications Menu	8
1.4.3	No Desktop Icon	8
2	Using Recall	8
2.1	Basic Operation	8
2.1.1	Opening a File upon Start	8
2.1.2	Using a Flashcard List	9
2.1.3	Opening a Different File	10
2.1.4	Exiting	10
2.1.5	Getting Help	10
2.2	Creating a Flashcard File	11
2.2.1	Creating a File	11
2.2.2	Formatting Question/Answer files	11
3	Contributing	11
3.1	General Information	11
3.1.1	GitHub Repository	11
3.1.2	Types of Contributions	11
3.1.3	Languages and Libraries Used	11
3.2	Contributing to the Program	11
3.2.1	Bug Fixes and Minor Changes	11
3.2.2	New Features and Major Changes	11
3.2.3	Compiling Recall	12
3.2.4	Creating an Installer Zip	12
3.2.5	Other Make Commands	12

3.3	Contributing to Documentation	12
3.3.1	Contributing to the Recall Manual	12
3.4	Licence	12

1 Installation

1.1 Graphical Installation – Simplest

1.1.1 Downloading Installer

Begin by downloading the appropriate `recall_installer.zip` file for your operating system. All variants can be found on the Recall Downloads page: https://various-and-sundry.com/recall_downloads.html.

1.1.2 Unzipping Installer

Navigate to the `recall_installer.zip` file (example shown in figure 1).

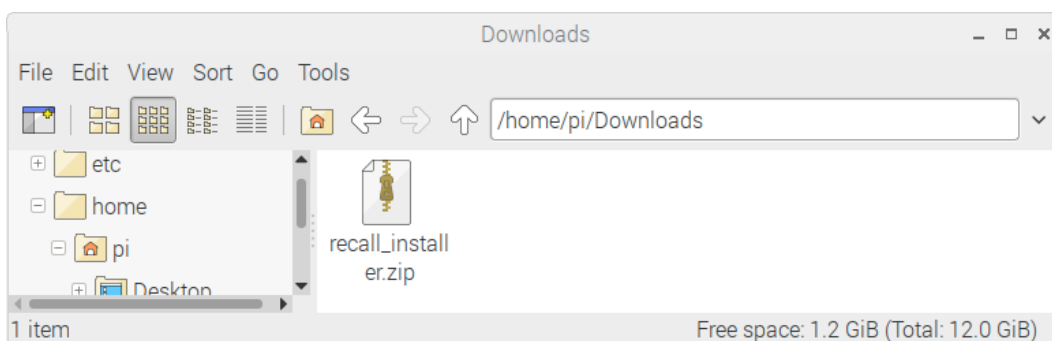


Figure 1: *recall_installer.zip*.

Unzip the *recall_installer.zip* file. In most desktop environments, clicking on the file with the right mouse button will display an menu containing the option 'Extract Here' (example shown in figure 2). If no 'Extract Here' option is available, the file must be extracted via another method. There may be an 'Extract To...' option, or the use of an archive program may be necessary.

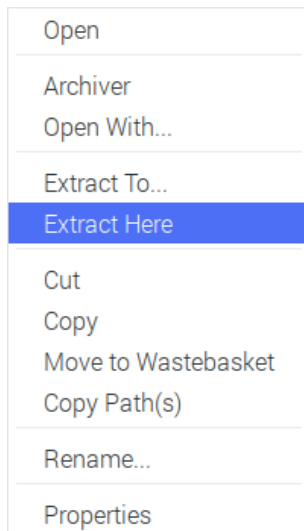


Figure 2: An example menu displayed after right clicking the *recall_installer.zip*.

When the contents of *recall_installer.zip* have been extracted, a directory named *recall_installer* will be created (shown in figure 3). Click on this new directory to view its contents.

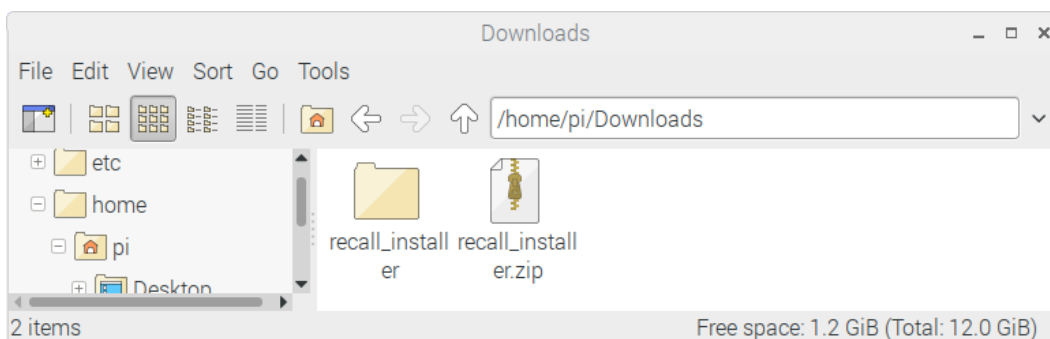


Figure 3: Extracted *recall_installer* directory alongside *recall_installer.zip*.

1.1.3 Running Install Script

Within the extracted *recall_installer* directory, several files will be present (as seen in figure 4). These include the install and uninstall scripts.

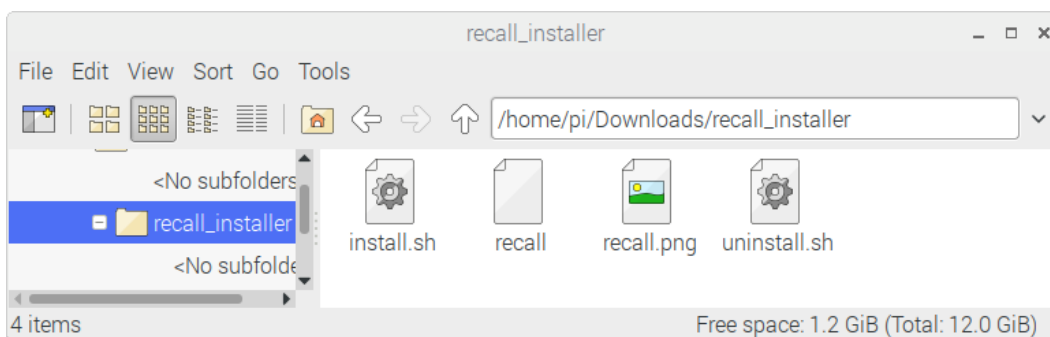


Figure 4: Contents of the *recall_installer* directory.

If the *install.sh* file is clicked with the right mouse button, a menu should be displayed (an example is shown in figure 5). This menu should display an option to run the file (*open*, *run*, *execute*, *etc*).

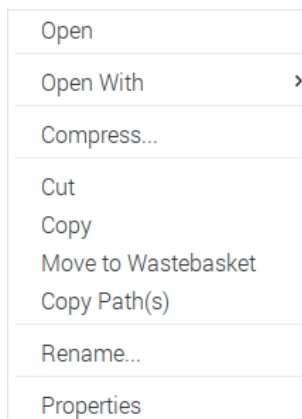


Figure 5: Example menu displayed after right clicking *install.sh*.

After clicking the appropriate option, the script may execute or a prompt such as the one shown in figure 6 may appear. Select 'Execute' or 'Execute in Terminal' to run the install script.

This script will install Recall in the *~/.recall* directory which will be created in the home directory. If the directory *~/Documents* exists, the installer will create *~/Documents/RecallLists* to store the flashcard files. If there is no *~/Documents* directory, the installer will create *~/.recall/RecallLists* to store the flashcard files. A Recall launcher will be added to both the applications menu and the desktop. The *~/.recall* will also be added to the

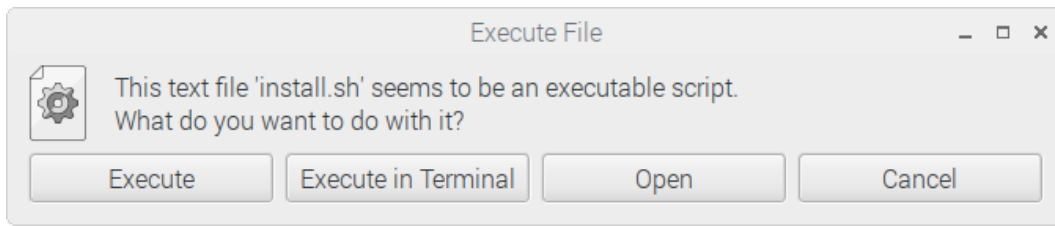


Figure 6: Example of execution prompt.

`$PATH` valuable, allowing Recall to be invoked via the command *recall* (this may not work until the shell has been restarted).

1.1.4 Potential Problems

On some desktop environments, it may be difficult to run the install script graphically. The script may open as a plain text file instead of executing. In this case, it may be best to run the file from the command-line. Navigate into the *recall_installer* directory from the command-line and follow the instructions in section 1.2.3.

The install script may not work as expected on all desktop environments and/or GNU/Linux distributions. Known issues are explained in section 1.4.

1.2 Command-Line Installation – Most Reliable

1.2.1 Downloading Installer

Begin by downloading the appropriate installer *.zip* file for your operating system. The web addresses of the three available *.zip* installers are shown below.

https://various-and-sundry.com/downloads/recall/recall_installer_i686_ncurses5.zip

https://various-and-sundry.com/downloads/recall/recall_installer_i686_ncurses6.zip

https://various-and-sundry.com/downloads/recall/recall_installer_x86-64_ncurses6.zip

The files can be downloaded via the *wget* command:

wget https://various-and-sundry.com/downloads/recall/name_of_file.zip

This command will download the installer *.zip* into the current working directory (as shown in figure 7).

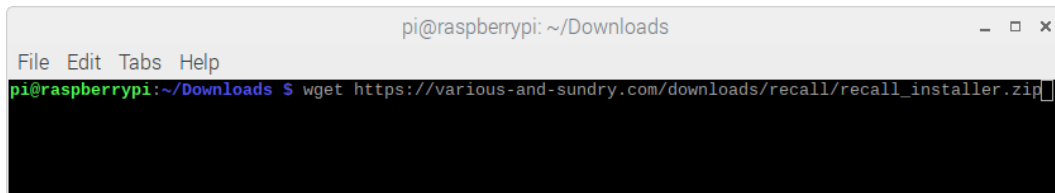


Figure 7: Wget downloading *recall_installer.zip*.

Wget should produce a file named *recall_installer.zip* as shown in figure 8.

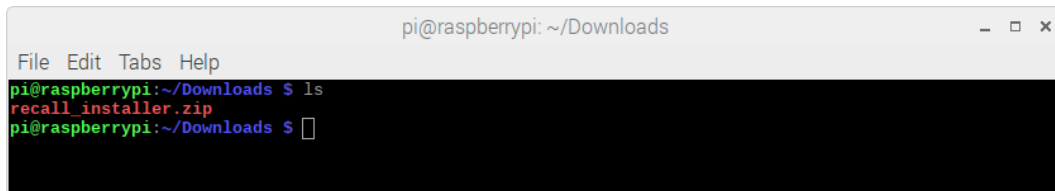


Figure 8: Downloaded installer file.

1.2.2 Unzipping Installer

Now the installer must be unzipped. The *unzip* command can be used to accomplish this. Run the command *unzip recall_installer.zip* as shown in figure 9.

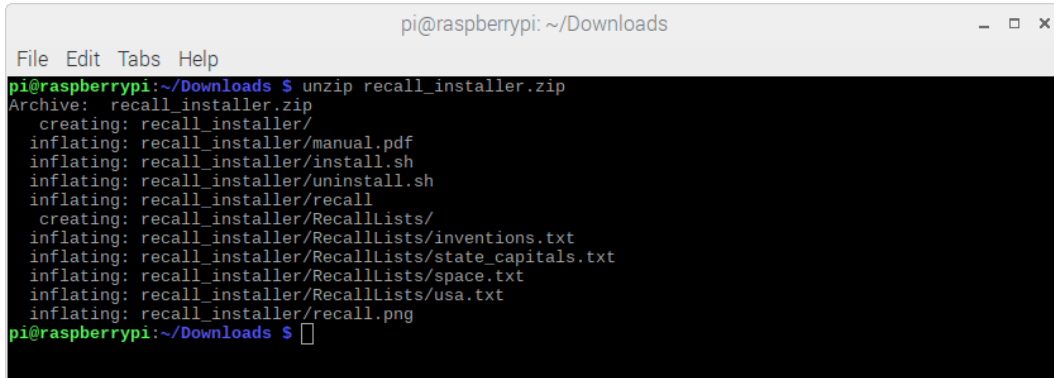
A terminal window titled 'pi@raspberrypi: ~/Downloads' with a menu bar (File, Edit, Tabs, Help). The command 'unzip recall_installer.zip' has been executed, showing the following output: 'Archive: recall_installer.zip', 'creating: recall_installer/', 'inflating: recall_installer/manual.pdf', 'inflating: recall_installer/install.sh', 'inflating: recall_installer/uninstall.sh', 'inflating: recall_installer/recall', 'creating: recall_installer/RecallLists/', 'inflating: recall_installer/RecallLists/inventions.txt', 'inflating: recall_installer/RecallLists/state_capitals.txt', 'inflating: recall_installer/RecallLists/space.txt', 'inflating: recall_installer/RecallLists/usa.txt', 'inflating: recall_installer/recall.png'. The prompt is now 'pi@raspberrypi:~/Downloads \$'.

Figure 9: Unzipping *recall_installer.zip*.

When the extraction is finished, a directory named *recall_installer* will have been created. After running *ls*, the *recall_installer* directory should be seen along with the *recall_installer.zip* (as shown in figure 10).

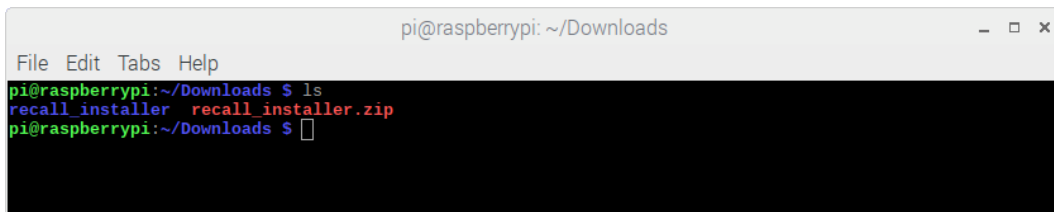
A terminal window titled 'pi@raspberrypi: ~/Downloads' with a menu bar (File, Edit, Tabs, Help). The command 'ls' has been executed, showing the output: 'recall_installer recall_installer.zip'. The prompt is now 'pi@raspberrypi:~/Downloads \$'.

Figure 10: Extracted directory along with installer zip.

Now enter the *recall_installer* directory with the command *cd recall_installer* (as shown in figure 11).

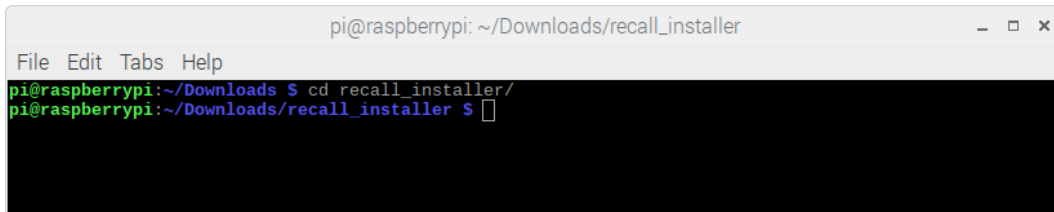
A terminal window titled 'pi@raspberrypi: ~/Downloads/recall_installer' with a menu bar (File, Edit, Tabs, Help). The command 'cd recall_installer/' has been executed, and the prompt has changed to 'pi@raspberrypi:~/Downloads/recall_installer \$'.

Figure 11: Entering *recall_installer* directory.

If the command *ls* is executed, the files and directories within *recall_installer* will be visible as shown in figure 12. These files include the install script.

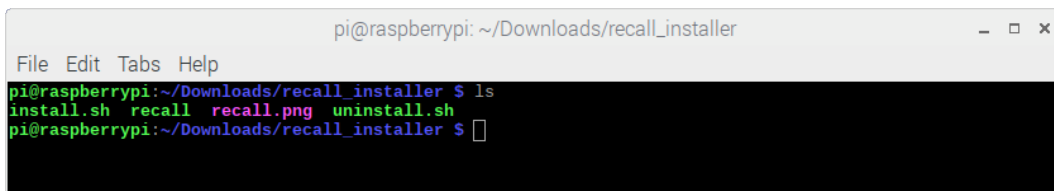
A terminal window titled 'pi@raspberrypi: ~/Downloads/recall_installer' with a menu bar (File, Edit, Tabs, Help). The command 'ls' has been executed, showing the output: 'install.sh recall recall.png uninstall.sh'. The prompt is now 'pi@raspberrypi:~/Downloads/recall_installer \$'.

Figure 12: Files within *recall_installer* directory.

1.2.3 Running Install Script

Now the install script (*install.sh*) can be invoked. Bash can be directly invoke upon it by running the command *bash install.sh* (as shown in figure 13). No errors were encountered if the script completes without producing any console output. The script will have installed Recall in a hidden directory in the home directory (*~.recall*). It will also create a directory called *RecallLists* in which to store flashcard files. If the directory *~/Documents* exists, the installer will create *~/Documents/RecallLists* to store the flashcard files. If there is no *~/Documents* directory, the installer will create *~/.recall/RecallLists* to store the flashcard files. A graphical launcher icon will also be added to the launch menu and the desktop. The *~/.recall* directory will also be added to *\$PATH* variable, allowing Recall to be invoked by typing *recall* into the command-line (this may not work until the shall has been restarted).

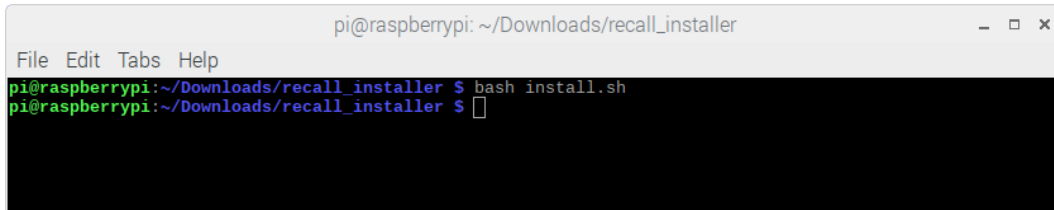
A terminal window titled 'pi@raspberrypi: ~/Downloads/recall_installer'. The prompt is 'pi@raspberrypi:~/Downloads/recall_installer \$'. The user has entered 'bash install.sh' and the prompt is now 'pi@raspberrypi:~/Downloads/recall_installer \$' with a cursor.

Figure 13: Running *install.sh*.

If recall is typed into a command-line, shown in figure 14, Recall should start as seen in figure 15. This command must be invoked in a shall that was starded after the install script had finished. That is because the changes that the install script made to the *\$PATH* will only be loaded in shells that were started after the install script made the changes.

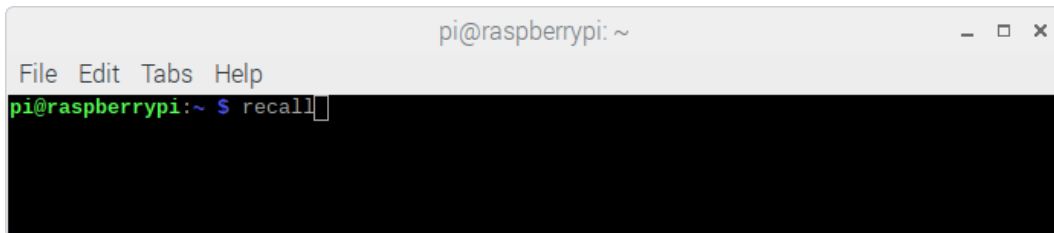
A terminal window titled 'pi@raspberrypi: ~'. The prompt is 'pi@raspberrypi:~ \$'. The user has entered 'recall' and the prompt is now 'pi@raspberrypi:~ \$' with a cursor.

Figure 14: Invoking Recall by typing *recall*.

A terminal window titled 'pi@raspberrypi: ~'. The prompt is 'pi@raspberrypi:~ \$'. The user has entered 'recall' and the prompt is now 'pi@raspberrypi:~ \$' with a cursor. The application has started and displays a ASCII art logo for 'a memory training tool'. Below the logo, it says 'Type the name of the file you would like to open.' and there is a cursor.

Figure 15: Recall running.

1.2.4 potential problems

The install script may not work as expected in all desktop environments or on all GUN/Linux distributions. Potential issues are explained in section 1.4.

1.3 Manual Install – Not Recommended

1.3.1 Why Manually Install?

Recall can be installed manually, allowing for a customized installation. This will allow the directory in which the program is stored, the launcher(s) configuration, and the *\$PATH* to be manually customized. If there is no specific reason to modify these options, Recall can much more easily be installed via its install script as shown in sections 1.1 – Graphical Installation or section 1.2 – Command-Line Installation.

1.3.2 Downloading Necessary Files

Begin by downloading the appropriate installer *.zip* file for your operating system. The web addresses of the three available *.zip* installers are shown below. They could also be download from the Recall Downloads page: https://various-and-sundry.com/recall_downloads.html.

https://various-and-sundry.com/downloads/recall/recall_installer_i686_ncurses5.zip

https://various-and-sundry.com/downloads/recall/recall_installer_i686_ncurses6.zip

https://various-and-sundry.com/downloads/recall/recall_installer_x86-64_ncurses6.zip

The files can be downloaded via the *wget* command or by another means.

1.3.3 Storing Executable

Create a directory in which the program will be stored. By default, this would be *~/recall*, but any location will work. Copy *recall* and *recall.png* into this directory. *Recall* is the executable, and *recall.png* is the icon.

The icon is not necessary for the function of Recall. It can be excluded or replaced with a different icon. If the icon is omitted, the recall logo will not appear on the program's launcher. This will not effect the program's functionality, and if you are not using a graphical environment, it will make no difference whatsoever.

1.3.4 Adding Recall to \$PATH

In order to run Recall from the shell, the path to the executable must be added to the *\$PATH* variable. This is done by editing *~/bashrc*. If recall is stored in the *~/recall* directory, the following line should be added to *~/bashrc*.

```
export PATH=$PATH:~/recall
```

If the executable is stored in a different directory, this line must be changed accordingly.

1.3.5 Creating Launchers

This step is optional, because Recall can be launched via the command-line, but it is often useful to have a graphical launch icon.

Create a blank file with a *.desktop* file extension. Under most circumstances, it makes sense to name this file *Recall.desktop*, but any file name will function as long as it has a *.desktop* file extension.

Within that file, add the following lines:

```
[Desktop Entry]

Type=Application
Name=Recall
Comment=A virtual flashcard tool.
Exec=bash $HOME/.recall/recall
Terminal=true
Icon=$HOME/.recall/recall.png
Categories=Education
```

These options can be modified as needed. The *Exec* option will need to be modified if the executable is stored in a directory other than *\$HOME/.recall* or if the executable has been named something other than *recall*. The *Icon* option can also be changed if the location or name of the icon have been changed.

The launcher can be placed on the desktop to provide a desktop launch icon.

To add the launcher to the application launch menu, copy it into the */usr/share/applications* directory. In most desktop environments, this will add the launcher to the applications menu.

1.4 Why the Install Script isn't Working

1.4.1 Recall Will not Run when Invoked from the Command-Line

If Recall's launchers work, but it will not run via a shell command (i.e. *recall*), the *\$PATH* variable may not include the directory in which Recall is installed (by default, *~/recall*). This problem is likely caused by the install script failing to modify the *~/bashrc* file correctly. Make sure that *~/recall* has been added to the *\$PATH* variable.

1.4.2 Recall was not added to the Applications Menu

This install script needs to be run by a user with sudo permission in order to add a launcher to the applications menu (i.e. */usr/share/applications*). If this permission is not provided, the install script will not be able to add a launcher to the applications menu, but it should still install recall, add it to the *\$PATH*, and create a desktop launcher.

1.4.3 No Desktop Icon

The install script will attempt to create a desktop launcher icon by adding one to the *~/Desktop* directory. If this file does not exist, then this action will fail, but the rest of the installation should be successful. If the *~/Desktop* directory exists but the desktop environment does not support desktop icons, then there will be no visible desktop icon.

2 Using Recall

2.1 Basic Operation

2.1.1 Opening a File upon Start

When Recall is launched, it will prompt to user to enter the name of a file (as shown in figure 16).

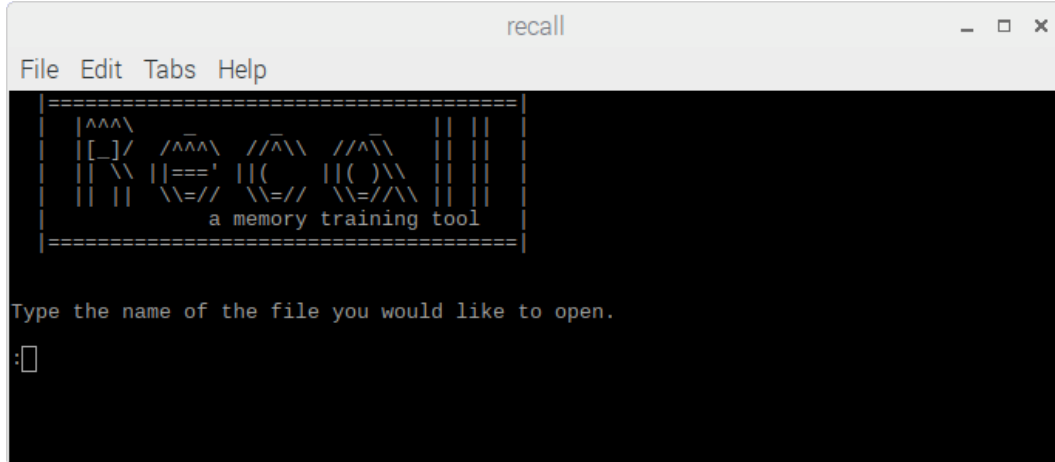


Figure 16: Recall prompting user to enter file name.

Flashcard files must be stored in a specific directory. By default, this directory will either be *~/Documents/RecallLists* or *~/recall/RecallLists*. This directory can be changed to any other directory. Recall will access whatever directory is given in the *PATH* line of the *~/recall/recallrc* file. For example, if the flashcard files are stored in the *~/Documents/RecallLists* directory, the *~/recall/recallrc* file should contain the following line.

```
PATH ~/Documents/RecallLists
```

The files that contain questions and answers must have a *.txt* file extension. To open one of these files, enter its name, without the *.txt* extension, into Recall's file prompt. Recall will not find the file if the *.txt* extension is included. Figure 17 shows a file name entered into the prompt.

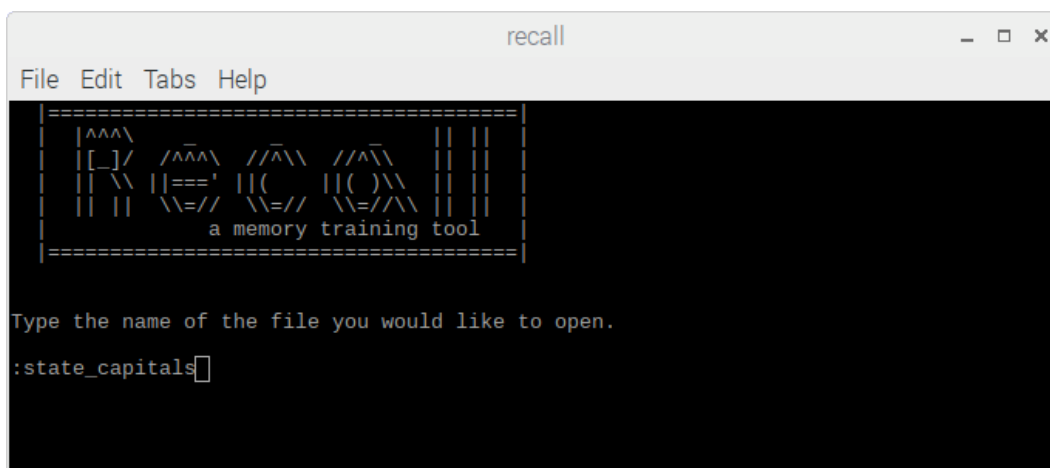


Figure 17: Opening *state_capitals.txt*.

2.1.2 Using a Flashcard List

Once a file has been opened in Recall, a random question from that file will be displayed (shown in figure 18). The user can then consider the question and attempt to recall the answer.



Figure 18: Example Question.

Any key can be pressed to reveal the answer (as shown in figure 19). Some keys such as *Home*, *End*, *Page Up*, *Page Down*, *arrows keys*, and *Insert* will yield unwanted behavior, and they should be avoided.

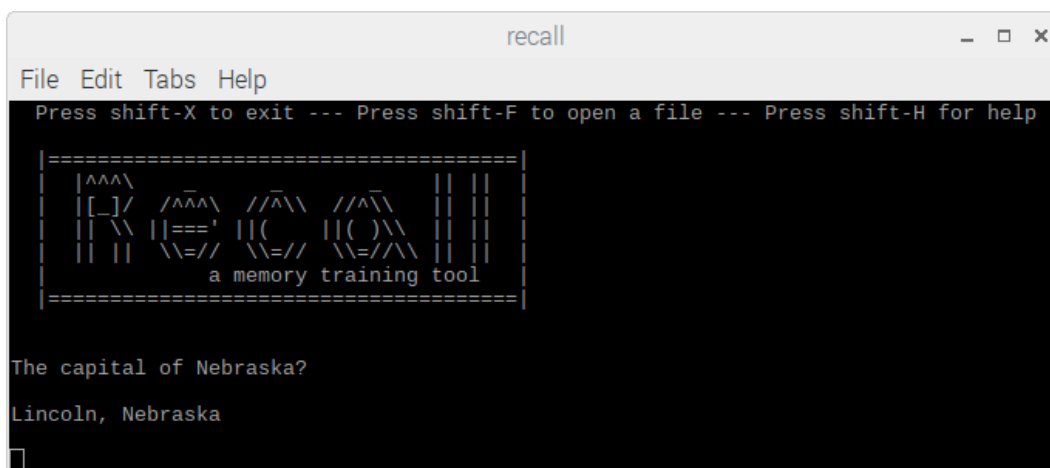


Figure 19: Example Question with Answer

Then, any key can be pressed to show another random question (shown in figure 20). Questions and answers can be cycled as quickly or slowly as the user desires.

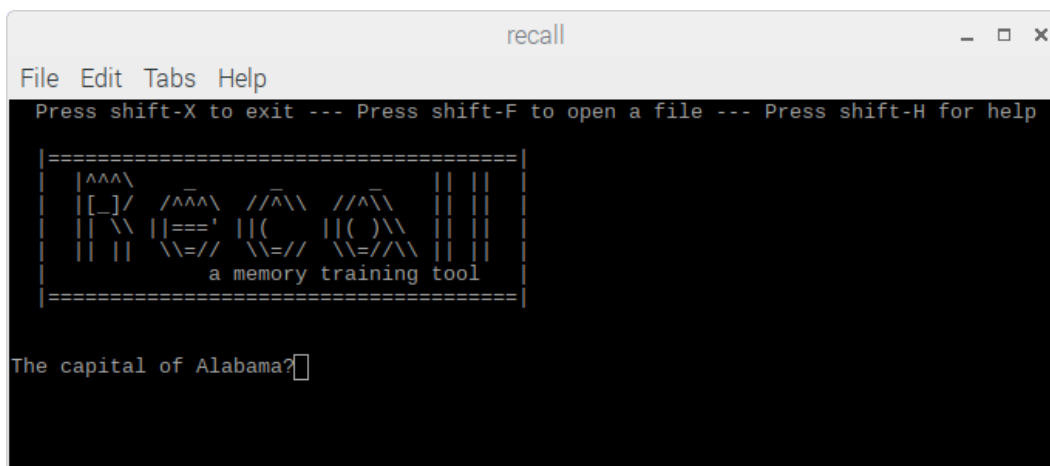


Figure 20: Another Example Question.

2.1.3 Opening a Different File

When cycling through questions, pressing shift-*f* (capital *F*) will bring the user back to the file selection prompt. That allows the user to select a different flashcard file. The instructions in section 2.1.1 explain the process of selecting a file.

2.1.4 Exiting

When cycling through files, shift-*x* (capital *X*) will close Recall. Because Recall is a command-line application, Ctrl-*c* will also close Recall. When Recall is used in a graphical environment, Recall can also be closed by closing the graphical window in which it is displayed.

2.1.5 Getting Help

When cycling through questions, a help menu can be accessed by pressing shift-*h* (capital *H*). Pressing any key will then exit the help menu.

2.2 Creating a Flashcard File

2.2.1 Creating a File

All flashcard files must be stored in the directory specified by the *PATH* line of the *~/.recall/recallrc* file. By default, this directory is either *~/Documents/RecallLists* or *~/.recall/RecallLists* directory (depending on where *~/Documents* exists). All flashcard files must also have a *.txt* file extension. Any file that fulfils these requirements will be recognised by Recall and can be opened with *open file* prompt by typing its name without the file extension.

2.2.2 Formatting Question/Answer files

Question and answer pairs can be added to and removed from any Recall file with a text editor. Each question/answer pair must be listed on its own line. Each question must be at the beginning of a line, and the answer must follow on the same line. They must be separated by a grave accent (`'`). Below is an example of three correctly formatted lines of a flashcard file.

```
What is the first letter of the alphabet?'The first letter of the alphabet is 'A'.
What is the second letter of the alphabet?'The second letter of the alphabet is 'B'.
What is the third letter of the alphabet?'The third letter of the alphabet is 'C'.
```

Any number of lines can be added to or removed from a flashcard file. Any lines that do not contain a grave accent will be ignored by Recall. Any line without a grave accent is considered a comment. Empty lines will also be ignored by Recall.

3 Contributing

3.1 General Information

3.1.1 GitHub Repository

Recall is free software (section 3.4), and contributions and suggestions are welcome. The Recall GitHub repository can be found at <https://github.com/various-and-sundry/recall>.

3.1.2 Types of Contributions

Recall is a relatively simple non-graphical program, but many features can potentially be added. All suggestions are welcome. Additions to and clarification of documentation would also be appreciated. Porting Recall to other operating systems may also be beneficial.

3.1.3 Languages and Libraries Used

Recall is written entirely in C and uses the ncurses library. The manual is written in L^AT_EX.

3.2 Contributing to the Program

3.2.1 Bug Fixes and Minor Changes

If a minor bug or issue is found, anyone is welcome to correct it and make a pull request. If that is not convenient, an issue can be filed so that the maintainer can correct the issue.

3.2.2 New Features and Major Changes

If an individual would like to make a major addition to Recall, it is probably best that he/she first file an issue proposing the change. That will allow the individual and the maintainer to discuss the idea before any work is done. Additionally, the maintainer will be very excited to see anyone of any experience level interested in contributing.

3.2.3 Compiling Recall

Recall is currently compiled from only one source file: *recall.c*. The makefile will automatically compile Recall with the command *make* or *make recall*. If *make* is not installed, the command *gcc recall.c -o recall -lncurses* can be used to compile Recall.

Recall requires the *ncurses* library. When using Debian or most other Debian based systems, the developer's libraries for *ncurses* can be install via the following command.

```
sudo apt-get install libncurses5-dev libncursesw5-dev
```

When using Fedora, Red Hat, CentOS, or one of their derivatives, the developer's library for *ncurses* can be installed via the following command.

```
sudo yum install ncurses-devel
```

3.2.4 Creating an Installer Zip

The command *make recall_installer.zip* will automatically generate a zipped installer file. This installer file will contain all of the files necessary to install Recall including the install script. It is best practice to run *make clean* before running *make recall_installer.zip*. Otherwise, the command may encounter errors.

3.2.5 Other Make Commands

The command *make clean* will removed the compiled program, the installer zip, and temporary files that were created while the installer was created. The commands *make install* and *make uninstall* will run the *install.sh* and *uninstall.sh* scripts respectively.

3.3 Contributing to Documentation

3.3.1 Contributing to the Recall Manual

Within the Recall source repository, documentation exists in a L^AT_EX file (*manual.tex*). All improvements to the manual are welcome.

3.4 Licence

Recall is under the MIT Licence as seen below.

MIT License
Copyright (c) 2020 I. D. Markle

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.