HW 5: Programming, Document, and Graphical Tools

Due: December 5

Overview: This is an assignment designed to give you practical experience with three sets of tools covered in class (programming, graphical, and document creation tools). The programming tools utilized are `make` , `gprof` , `diff` , and the GNU C/C++ compiler. Graphviz, Imagemagick, and LaTeX/TikZ will be used for generating graphics in a document form.

Objective: Complete the following tasks (in the order presented) to visualize the execution of a small calendar tool called `remind`. Each problem will have one or more outputs, taking the form of a short answer (text), code, a PNG image, or a PDF. After completing each task, clearly specify which output is matched with each problem to receive credit.

1. Download the calendar utility `remind` with the URL

   `https://www.roaringpenguin.com/files/download/remind-03.01.15.tar.gz`

   and extract the contents. After running `./configure` at the root of the directory, find the `Makefile` in the `src` folder (not at the root). The file you're looking for should have this at the top:

```
# Makefile.in for REMIND
#

SHELL= /bin/sh
BETA = 1
srcdir=.
prefix=/usr/local
exec_prefix=${prefix}
mandir=${datarootdir}/man
bindir=${exec_prefix}/bin
datadir=${datarootdir}
datarootdir=${prefix}/share

VERSION=03.01.15

INSTALL=/usr/bin/install -c
INSTALL_PROGRAM=${INSTALL}
INSTALL_DATA=${INSTALL} -m 644
```

After making a backup of this file, edit the `Makefile` so that it can be profiled with `gprof`. Explain the reasoning behind your change. Report (in this HW) the unified format (`-u` flag) output of the `diff` utility.

2. Compile the code using using the `Makefile` of `remind-03.01.15` package. Report where the `remind` executable is located within the package's directory after this step. Explain

   (a) How you might find `remind` if you read the Makefile, and

   (b) in general how you can find the (non-phony) targets of the Makefile without reading the Makefile at all (which is not always easy to read).

   Note: You do not have to worry about installing any of the binaries produced (i.e. with `make install`).
   Hint: It might help to use a utility covered in the `week_3.pdf` slides.

3. Profile the `remind` executable using `gprof`. The command you will be profiling is

   ```
   ./remind ../examples/defs.rem
   ```

   Report the 10 functions with the *fewest* calls as recorded in `gmon.out`.

4. Use `gprof2dot` to produce a dot file. Report the contents of the dot file here, and compile it with `dot` to produce a PNG.

Note: If you wish to run this from `linprog` (recommended), you will need to install both `pip` (the Python package manager) and `gprof2dot` locally (because you do not have `sudo` privileges). This can be done using the following steps:

(a) Get the pip install script:

```
wget https://bootstrap.pypa.io/get-pip.py
```

(b) Instruct the install script to install locally (in `.local/bin`):

```
python get-pip.py --user
```

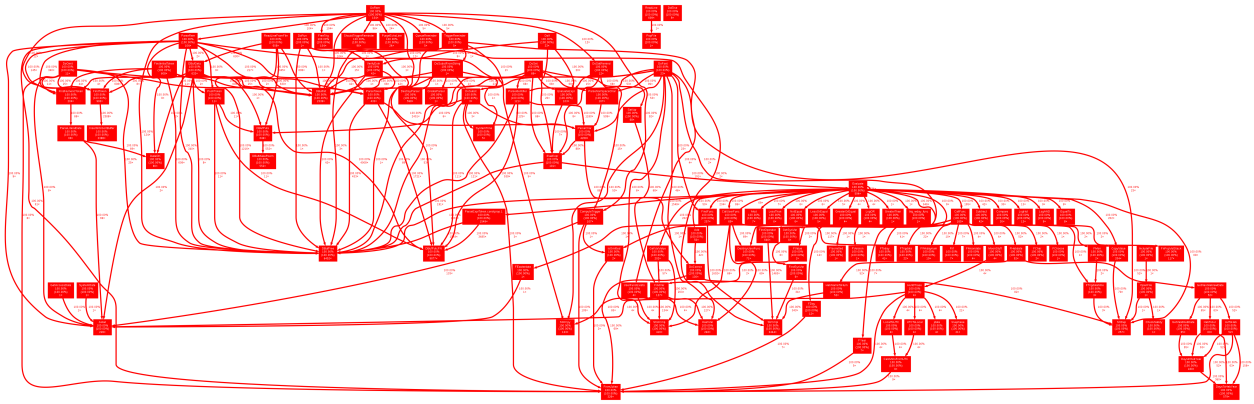
(c) Within `.local/bin` you will find `pip` . Install `gprof2dot` locally

```
./pip install gprof2dot --user
```

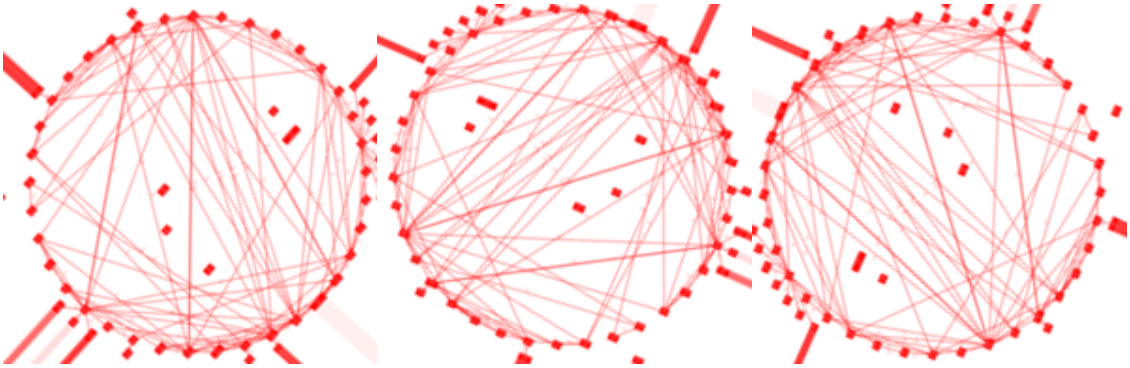
This will put the `gprof2dot` binary in the same directory as pip. Keep this in mind when using it (either by specifying its absolute path or by changing your `PATH` environment variable).

To copy images or other files produced from `linprog` , you might use WinSCP if you are on Windows or gFTP on Ubuntu (unless you are already comfortable with `scp` or `sftp`).

The rendered dot file (with `dot`) could look *roughly* as complex as this:



5. With TikZ, draw the root and first level of the tree produced by `gprof2dot`, and render it (as a PDF).
6. **Extra Credit** Use graphviz to create a circular call graph, and use Imagemagick to create a looping GIF of the call graph making a full rotation.



(a) 132° (b) 24° (c) 294°

Figure 1: Sample output of the circular call graph rotation

Hint: You will want to use Imagemagick's `-distort` flag with the `SRT` distortion. Don't use the `-rotate` flag!
Note: There will be some distortion around the edges with SRT; this is OK.

Submission:
Upload all the files in a single `tar` or `zip` file by the due date.