# POLLING SIMULATION PROGRAM

# USER GUIDE

**Introduction:**

To use this program, you must first supply two different files, a Configuration file and a Precinct File. The purpose of the configuration file is to specify limits the program needs to know. The first line of the configuration file needs a number of integers, separated by spaces, which are as followed; the first integer is a seed number, which is used for the randomizer. This number is used for a random number generator the code uses and needs to be changed every time the program is read. The second number is the length of the election day, or how long the polls are open for people to vote. This number is meant to be in hours. Next is the mean time to vote, which is the average time it took voters to cast their votes and leave the voting booth, which is in seconds. The next two numbers are the minimum and maximum voters that are expected and the number of voters needed to run the simulation. Next is the maximum wait time for the voter to cast their vote; as in, this number quantifies how long is too long to wait in order to vote. The last integer of the first line is the number of times the simulation needs to be run. The second line of the configuration file has percentages that account for the number of arrivals that occurred at the opening of the voting booths, as well as percentages for each following hour until the voting booth closes. These numbers are written as doubles, in a format such that 10.0 would be 10% of voters, and 13 numbers are required, from hour 0 to hour 12. The second file that needs to be given is the file that holds all the data for the various precincts. Each line of the file has data for one precinct, and the data is listed as follows; the first number is the Precinct number, which is an integer written in the format of '###'. The next value is the name of the Precinct, which is a string written in the format of 'XXX#####'. Following that is a double representing the turnout of voters for that specific precinct, which is written in the same way as the percent from the config. Next is an integer that is the number of voters that showed up at a specific precinct. Following that is an integer that represents the number of voters that is expected to show up at a precinct. Following that is the number of voting stations that are available for use at a precinct. Then comes another double which represents the Precinct minority. Lastly, there are three integers representing statistics used in the simulation, and should be set to 0 by default.

The program can then find and properly interpret these files once they are formatted properly, placed in the executing directory (with respects to the executable), and then have their respective names be supplied as command line arguments to the program itself. After this process is completed and the user runs the executable, with the command line arguments supplied. Once the program is finished running it will output useful information onto the supplied file via the command line arguments. This file contains all of your program output. First, it gives back all the information the user fed into the program through the configuration file, so the user will be able to make sure everything was entered correctly. The file also returns the maximum service time, which is the longest time in seconds that a voter had to vote. The output file then lists out the percentages of the arrivals from each hour, which was given in the precinct file, and a compares numbers to the wait time that was listed as "too long."

**Formatting Your Files:**

**Config File:**

The first line should be these values separated by spaces:

- Seed for Random Number Generator. Should be changed each use.
- Length of the election day (in hours).
- Mean time for one vote to be cast (in seconds).
- Minimum number of voters needed to carry out a simulation.
- Maximum number of voters needed to carry out a simulation.
- Wait time that is too long (in minutes).
- Number of times to run the simulation.

**Example: 35 13  105 50 5000 30 3**

The second line should be these values separated by spaces:

- Percent of arrivals at opening time (enter 10.0 for 10%).
- Percent of arrivals at each hour of the day (enter 10.0 for 10%).

**Example: 0.0 10.0 10.0 10.0  5.0  5.0  5.0 10.0 10.0  5.0  5.0  5.0 10.0 10.0**

**Precinct File:**

Each line should have this precinct data, in this order and separated by spaces, for each precinct.

- Precinct number.
- Precinct name.
- Precinct turnout (percentage of total voters).
- Number of voters.
- Expected number of voters.
- Number of stations.
- Precinct minority.

**Example: 000 XXX00000 19.2 10101 0 235 8 10.1 0 0 0**

(Notice the 3 trailing 0's. These should be 0 as a default.)

**Reading Output:**

- The input values given in the config file.

- Max service time subscript, the longest time a voter had to wait (in seconds).

- The percentages of arrivals at each hour of the day.

- Compare to the "too long" wait time to see if a voter had to wait too long.

- The percentages of arrivals at each hour of the day (as given in the precinct file).

**Executing the program:**

Once the required input files have been created in the main directory, executing the program is simple. To execute the program, navigate to the directory using terminal. Once you have done this, type 'make' in the command line and press enter. This will essentially 'create' the program and make it executable. Finally, to execute the program, type the following and press enter:

'**./Aprog config_filename pct_filename output_filename log_filename**'

**(Make sure to use your .txt filenames.)**


**User Guide Authors:**

C.J. Waldron

Abraham Khan

Stephen Page

Thomas O'Hara

**Program Author:**

Duncan Buell


**Finalized 1 December 2016.**