RUN IRV

Name	Run IRV
ID	UC_001
Description	An election official or a tester wants to run IRV on a collection of ballot votes
Actors	Election officials, testers
Organizational Benefits	Produces the results of an IRV with given information
Frequency of Use	Every time an IRV election file is run with the program.
Triggers	The actor runs the program with a file.
Preconditions	1. Every ballot has at least one candidate selected 2. Every ballot has at least two candidates
Postconditions	1. If there are two losers, then a coin must be flipped in order to determine the loser 2. Results get outputted to the screen
Main Course	1. Check file for initial header data (IRV/OPL, number of votes, number of candidates, candidate names) 2. Load file data into the heap 3. Shuffle the ballots 4. Sort ballots into corresponding lists for each candidate based on their first choice 5. Log # of votes for each candidate 6. Distribute votes of the lowest candidate to their next choice 7. Go to step 5 until there is a candidate with the majority vote 8. Output audit file and results to screen

Alternate Courses	AC1 file header reads "OPL"
	   1. Run OPL (UC_002) instead
	AC2 Shuffle is disabled (UC_004)
	   1. Do not shuffle ballots
	   2. Continue from MC4
	AC5 If a candidate gets >50% of the ballot
	   1. Declare winner
	   2. Go to MC 8
	AC6 Tie for lowest candidate
	   1. Flip a coin if enabled to determine lowest votes
	AC6 Ballot has no next choice
	   1. Remove ballot from the heap

Exceptions	EX. System fails to determine a winner a. Go back to step 1

RUN OPL

Name	Run OPL
ID	UC_002
Description	An election official or a tester wants to run OPL on a collection of ballot votes
Actors	Election officials, testers
Organizational Benefits	Produces the results of an OPL with given information
Frequency of Use	Every time an OPL election file is run with the program.
Triggers	The actor runs the program with a file.
Preconditions	1. Every ballot has exactly one candidate selected
Postconditions	1. If there candidates are tied for the remainder votes, do a coin flip to determine which party gets the seat 2. Results get outputted to the screen
Main Course	1. Calculate quota for advancement 2. Sort the party candidates by total votes 3. Advance the number of most popular candidates from each party equal to the floor of their total votes divided by the quota, saving the remainder. 4. For the remaining seats, the parties with the highest remainder have their candidates advanced. 5. Output audit file 6. Output the results to screen
Alternate Courses	AC2a. file header reads "IRV" 1. Set up a data structure to fill with IRV ballots. 2. Fill the data structure with ballots. 3. Run IRV (UC_001) AC4 Tie for Remaining Votes 1. Flip a coin to determine who gets the last seat.
Exceptions	EX. Fails to output winner to screen 1. System notifies that an error has occurred. EX. There are more seats than candidates available 1. System notifies that there are not enough candidates

CSV INPUT

Name	CSV Input
ID	UC_003
Description	An election official or tester wants to input the election data into the program as a CSV file
Actors	Election officials, testers
Organizational Benefits	Allows for the input of data to run elections
Frequency of Use	Every time an election is run
Triggers	The actor enters the terminal command with a CSV file as an argument
Preconditions	1. A CSV file has been created accurately with a header that lists election type, number of candidates, number of ballots, candidate names, and number of seats.
Postconditions	1. Ballots are loaded into the program and ready for processing
Main Course	1. Access file and create file input stream. 2. Check header for election type. 3. File header has no specified type 4. Ask actor for election type

Alternate Courses	AC2a. file header reads "IRV"
	   1. Set up a data structure to fill with IRV ballots.
	   2. Fill the data structure with ballots.
	   3. Run IRV (UC_001)
	AC2b. file header reads "OPL"
	   1. Set up a data structure to fill with OPL ballots.
	   2. Fill the data structure with ballots.
	   3. Run OPL (UC_002)

Exceptions	EX. Cannot access file and create file input stream(error/bug).

TOGGLE BALLOT SHUFFLE FOR IRV

Name	Turn off Ballot Shuffle for IRV
ID	UC_004
Description	A tester needs to be able to disable ballot shuffling for testing purposes.
Actors	Testers
Organizational Benefits	Allows for consistent testing of IRV ballot data
Frequency of Use	Any time tester wants to test IRV data
Triggers	Runs election program with shuffle-off flag
Preconditions	1. IRV ballot is being used.
Postconditions	1. Shuffle is turned off for the following IRV run (UC_001)
Main Course	1. Disable shuffling functionality of the IRV ballot counting algorithm
Alternate Courses	
Exceptions	EX. System gives an error when actor tries to turn off shuffle for an OPL ballot file

VIEW ELECTION RESULTS

Name	View election results
ID	UC_005
Description	Media personnel want to know the winner(s) and information about the election (e.g. type of election, number of seats).
Actors	Media officials
Organizational Benefits	The media knows the results of the election
Frequency of Use	Whenever election is completed
Triggers	IRV or OPL completed
Preconditions	All of the ballots have been collected and the winners have been determined.
Postconditions	Output file for simple, understandable election results
Main Course	1. File with tables are outputted showing the finals results, total votes, and percentage of votes for each candidate
Alternate Courses	
Exceptions	EX. Statistically incorrect results are obtained.

GENERATE AUDIT FILE

Name	Generate Audit File
ID	UC_006
Description	A file of election information and results for election officials.
Actors	Election Officials
Organizational Benefits	The election officials will be able to review the election results in case they expect an error or invalid election.
Frequency of Use	Whenever an election is completed
Triggers	IRV or OPL completed
Preconditions	All of the ballots have been collected and the winners have been determined.
Postconditions	Output file with all the necessary information to faithfully recreate the election.
Main Course	1. The system runs an election 2. Audit file is generated upon completion of the election
Alternate Courses	
Exceptions	EX. The system generates an audit file with inconsistent election results