Candidate - name:String - party: char - totalVotes:int - IRBallotList:ArrayList<IRBallot> + Candidate(name:string,party:char) + getName():String + getParty():char + addVote():void + getVoteCount():int + addBallotToList(irBallot:IRBallot):void + getIRBallotList():ArrayList<IRBallot> **IRBallot**

- rankedCandidateNames:string∏ - rankedCandidateParties:char[]
- rankedCandidateIndices:int[]
- topPickIndex:int = 0
- -id:int
- + IRBallot(ballotString:String, candidateParties: char[], candidateNamesOnly: String[], id: int):void
- + getTopPick():String
- + getTopPickParty():char
- + setNewTopPick(eliminatedCandidates:Candidates[]):void
- + getID():int

Ballot Counting System (Main)

- static electionType: int

- static ballotFile: File

- static ballotReader: Scanner

- static auditFile: File

- static auditFileWriter: FileWriter - static auditFileName: String

+ static main(args:String[]): void

+ static ir(): void

+ static getIRCandidateParties(candidateWithParty:

String[]): void

+ static generateIRReport(numCandidates: int, numBallots: int, winner: Candidate,

availableCandidates: ArrayList<Candidate>, eliminatedCandidates: ArrayList<Candidate>): void

+ static printResults(availableCandidates: ArrayList<Candidate>, eliminatedCandidates: ArrayList<Candidate>,winner: Candidate): void

+ static distributeIRVotes(irBallotList:

ArrayList<IRBallot>, availableCandidates:

ArrayList<Candidate>): void

+ static opl(): void

+ static createParties(candidateInfo: String[], parties:

ArrayList<Party>): void

+ static createAndDistributeOPLBallots(ballots:

OPLBallot[], candidateInfo: String[], parties:

ArrayList<Party>): void

+ static distributeSeatsToParties(numBallots: int, numSeats: int, parties: ArrayList<Party>): void

+ static getDateTime(): String

+ static createAuditFile(): void

+ static coinFlip(): int

- numSeatsToDistribute:int

+ Party(partyName:char)

- + addCandidate(candidate:Candidate):void
- + getCandidateList():ArrayList()<Candidate>
- + incrementPartyVoteCount(votrs: int): void
- + setPartyVoteCount(votes:int):void

- candidates: ArrayList()<Candidate>

- + getPartyVoteCount():int
- + getPartyName():char

partyVoteCount:int

- partyName: char

- + setNumSeatsToDistribute(numSeats:int): String
- + distributeSeatsToCandidates():void
- + getCandidatesGivenSeats():ArrayList<Candidate>

Party

- candidatesGivenSeats: ArrayList()<Candidate>

Report

-electionType: String -numCandidates: int -numBallots: int -winner: String

+ Report(electionType: String, numCandidates: int, numBallots: int, winner: String): void

+ printReport():void

+ generateReportFile(dateTime: String, candidateStandings:

String):void

OPLBallot

- selectedCandidate:string
- selecedParty:char
- -id:int
- +OPLBallot(ballotString:String, candidateList:String[], newID:int):void
- + getCandidate():String
- + getParty(): String
- + getID():int