a) Obstacles:

I initially was trying to look through pollString as one entire entity to check if it was valid. I decided to change this into looking at each individual state forecast as they were separated by commas.

For converting integers to characters, you have to subtract the ‘0’ character from the number character to find the actual number integer. Also, you must find which place the number is in is because a 3 character might actually be 30.

b) design of code:

isValidPollString

Adds comma to the end to make each state forecast consistent in format : (forecast,)

Uses commas to set initial forecast bound. Checks if first forecast is valid and removes it from the pollstring. Process repeats until poll string is length 0. If isValidStateForecast fails, the poll string is said to be false.

isValidStateForecast

Minimum length of 2, otherwise not valid.

Checks if state name is valid. If so, removes it from the forecast. VA4D -> 4D.

If forecast is now an empty string, it was a valid forecast.

Confirms that last character is a letter and first one is a digit.

Confirms that no 2 letter characters are in a row.

Confirms that every character is either a digit or a letter.

If anything fails, returns false. Otherwise returns true.

countSeats

checks to see if valid pollstring, else returns 1

checks to see if party is a letter, else returns false.

capitalizes party to match pollstring.

iterates through poll string looking for party character, ignoring if there is not a digit character before it(meaning it was part of the state code)

checks digits before, and for every digit it is before the party character is, adds it and multiplies it by the next power of 10 to account for digit places

to make sure the character value is equal to the integer value, it is subtracted by the 0 character

capitalizer – capitalizes every character in the string to ease comparisons. this was applied to every poll string before looking at it.

c) test cases:

isValidPollString

valid poll strings

VA

VA3D

VA3D4R67I

VA3D4R67I,PA8R13D,NE5I6D21R

VA,CT4R

invalid

3D4R

VA, CT4R

VG3D4R67I,PA8R13D,NE5I6D21R

VA 3D4R67I,PA8R13D,NE5I6D21R

V4R,CT5D

$DE4R

DER4

countSeats

VG3D4R67I,PA8R13D,NE5I6D21R d 0

should return 1

VA3D4R67I,PA8R13D,NE5I6D21R % 0

should return 2

VA3D4R67I,PA8R13D,NE5I6D21R d -14

sets third value to 22 returns 0

VA3D4R67I,PA8R13D,NE5I6D21R q -14

sets third value to 0 returns 0