Name: Ruben Sam Thomas
Net ID: rst180005

Project 2 Pseudocode

Function: Auditorium Constructor
Parameters: String fileName

- Create the first row
- Int Row = 1 and char seat = 'A'
- If file has opened and has information in it
    - Read in ticket type of first seat
    - First = new Seat(row, seat, ticketType)
    - Set current pointer to first and tempLeft ptr to first
    - Read another line information from the file to get information about the row
    - For 0 to length of line of row information
        - currentSeat = new Seat(row, seat++, line[currIndex])
        - To set the seat to the left of the current seat
            - currentSeat.setLeft(tempLeft)
        - To set the seat to the right of the current seat
            - currentSeat.setRight(tempRight)
        - tempLeft = current
    - Seat tempUp = first
    - Seat tempSeat = first
    - Checking to see if the input file has more information
        - Get the next line of seat information from the file.
        - tempLeft = null
        - Seat = 'A'
        - Increment the number of rows by 1
        - Again iterate through the line with row information
            - For 0 to line.length
                - currentSeat = newSeat(row, seat++, line[currIndex])
                - currentSeat.setLeft(tempLeft)
                - currentSeat.setUp(tempUp)
                - tempSeat.setDown(current)
                - Checking to see if there is seat to the right
                    - if(currIndex != 0)
                        - tempLeft.setRight(current)
                    - tempLeft = current
                    - tempUp = tempUp.setRight()
            - firstSeat = firstSeat.getDown()
            - tempUp.firstSeat
    - totalRows = row
    - totalColumns = (int)(seat - 'A')

Name: Ruben Sam Thomas
Net ID: rst180005

Function Name: Input validation(Generic)<E>
Parameters: E userInput, E totalRowOrSeats, char type
Return type: boolean

- (If type is a row('R') then totalRowOrSeats would contain the total number of rows in the auditorium)
    - For loop to iterate through the total number of rows
        - If userInput is less than or equal to totalRowOrSeats
            - Return true
        - Else
            - Return false
- (If type is a seat('S') then totalRowOrSeats will contain the final alphabet of the seat in the auditorium)
    - For loop to iterate through the characters of the seats
        - If userInput is equal to any seat alphabet in totalRowOrSeats
            - Return true
        - Else
            - Return false

Function Name: Best Available
Parameters: auditorium, seatsRequired, totalSeats, totalRows
Return type: Seat

- Create different variable to use for the bestRow, bestColumn, bestDistance
- For loop to iterate through all the total rows in the auditorium
    - Another for loop to iterate through the total seats in each row
    - Check to see if the seats required by the user is present
        - If available
            - Calculate the distance of the current seat from the middle seat
            - Calculate the distance of the row from the middle row of the auditorium
            - If the distance calculated > tempDistance
                - Set bestDistance to tempDistance
                - Set the number of the bestRow
                - Seat the character of the best column
            - Else if distance calculated is equal to the tempDistance
                - If distance calculated is less than bestDistance
                    - Assign variables bestRow, bestColumn and bestDistance with the new values
                - Else if the distance calculated is equal to the bestDistance
                    - Check to see which row number is smaller

Name: Ruben Sam Thomas
Net ID: rst180005

- Set the variables bestRow, bestColumn and bestDistance with the new values
- Else
  - Return back and continue to check the other seats and rows
- Finally return the seat object to get the row and column of the best seat available