Project 2 Pseudocode

**Auditorium Constructor**

* Create ifstream file input object
* Ensure the file exists
  + Create pointers to 4 TheaterSeats and assign to null: up, down, left, and right
  + For loop until getline() hits an EOF
    - For loop through the lineStr until the null character is reached
      * Allocate a new TheaterSeat object and set its row, seat, reserved, and ticketType based upon what character is read from the lineStr and the increments in the for loops
      * If the increments in both for loops are zero, assign the first pointer variable in the Auditorium to the TheaterSeat just created
      * If the pointer to the up TheaterSeat is not null
        + Assign the current TheaterSeat’s up pointer to that up TheaterSeat
        + Assign the down pointer of the up TheaterSeat to the current TheaterSeat
        + Move the TheaterSeat up pointer variable to the one to the right of it (could be null)
      * Otherwise assign the current TheaterSeat object’s up pointer to null
      * If the pointer to the down TheaterSeat is not null
        + Assign the current TheaterSeat’s down pointer to that down TheaterSeat
        + Assign the up pointer of the down TheaterSeat to the current TheaterSeat
        + Move the TheaterSeat down pointer variable to the one to the right of it (could be null)
      * Otherwise assign the current TheaterSeat object’s down pointer to null
      * If the pointer to the left TheaterSeat is not null
        + Assign the current TheaterSeat’s left pointer to that left TheaterSeat
        + Assign the right pointer of the left TheaterSeat to the current TheaterSeat
        + Move the TheaterSeat left pointer variable to the one to the right of it (could be null)
      * Otherwise assign the current TheaterSeat object’s left pointer to null
      * If the pointer to the right TheaterSeat is not null
        + Assign the current TheaterSeat’s right pointer to that right TheaterSeat
        + Assign the left pointer of the right TheaterSeat to the current TheaterSeat
        + Move the TheaterSeat right pointer variable to the one to the right of it (could be null)
      * Otherwise assign the current TheaterSeat object’s right pointer to null
* If the file doesn’t exist, output and error and quit the program

**Auditorium Destructor**

* Create a pointer and assign it to the Auditorium’s first pointer
* For loop through the row until the current TheaterSeat pointer is null
  + For loop through the columns until the current TheaterSeat pointer is null
    - Create a temp TheaterSeat pointer and assign it to the current pointer’s right TheaterSeat
    - Free the current TheaterSeat
    - Assign the current pointer (now empty) to the temp variable
* Free the Auditorium

**Input Validation**

This will be a part of the reserve function which takes in the Auditorium

* Declare variables for ruw num, starting seat letter, num adult tickets, number of child tickets, and number of senior tickets
* While true
  + Prompt user for row number and store in string
  + If there is no error when converting to an int via stoi()
    - break
  + Otherwise
    - Output and error message
* While true
  + Prompt user for starting seat letter
  + If the starting seat letter is only one char long and is between the ASCII values for A and A + theater length
    - break
  + Otherwise
    - Output an error message
* While true
  + Prompt user for number of adult tickets
  + If there is no error when converting the number of adult tickets to an int and it is greater than or equal to 0 and is less than the length of the auditorium
    - break
  + Otherwise
    - Output an error message
* While true
  + Prompt user for number of child tickets
  + If there is no error when converting the number of child tickets to an int and it is greater than or equal to 0 and this+num adult tickets is less than the length of the auditorium
    - break
  + Otherwise
    - Output an error message
* While true
  + Prompt user for number of senior tickets
  + If there is no error when converting the number of senior tickets to an int and it is greater than or equal to 0 and this+num adult+num child tickets is less than the length of the auditorium
    - break
  + Otherwise
    - Output an error message

**Best Available**

Parameters: Auditorium, desired number of adult/child/senior tickets

Returns: best starting seat number (int)

This function already assumes that the desired row number and starting seat could not be fulfilled

* Initialize pointer to point to the first node in the auditorium
* Initialize minDistRow to the length of the row, and minDistCol to the length of the column
* Initialize bestSeatNum to -1 and bestRow to -1
* For loop that goes through rows of the auditorium
  + For loop that goes through the columns of the auditorium
    - Starting at the current pointer in the auditorium, run through each seat to the right of the current seat until the number of desired tickets was checked sequentially
      * If this fails
        + break
      * If this succeeds
        + Calculate the row and column distance
        + If the row distance is less than the minDistRow, update the minDistRow and bestRow
        + If the col distance is less than the minDistCol, update the minDistCol and bestSeatNum
* Decide on the best seat based upon which is closer to the middle of the auditorium, with the smaller row number taking precedence