# **Pseudo Code**

# Main.cpp

**Include header files**

**Main()**

**Declare class variables**

**Declare variables**

**Call functions in order**

**Pause**

**Return EXIT**

# Customer.h

**Customer::GetLogin**

**Prompt for and accept email and password**

**Store email and password**

**Increment store variable**

**IF not first customer**

**FOR counter goes from 0 to store variable**

**IF entered login equals any stored logins**

**Bool check variable equals true**

**Set first customer bool to false**

**Customer::GetProfile**

**Declare variables**

**IF check bool is true**

**Output “Welcome back customer”**

**Prompt for and accept verification code**

**ELSE**

**Prompt for and accept new profile info**

# Show.h

**Show::SelectShow**

**IF first time displaying shows**

**Output events list**

**Prompt for and accept customer’s desired show**

**IF not first time displaying shows**

**Call display show function**

**Set first time displaying shows bool variable to false**

**Show::DisplayShow**

**SWITCH (Integer variable)**

**CASE 1: Output ‘you have chosen first show’**

**CASE 2: Output ‘you have chosen second show’**

**CASE 3: Output ‘you have chosen third show’**

**CASE 4: Output ‘you have chosen fourth show’**

**DEFAULT: Output ‘Invalid input’, call to SelectShow function**

# ShowSeat.h

**ShowSeat::Initialise**

**FOR set up row width**

**FOR set up column length**

**Set A as value to fill field**

**ShowSeat::DisplaySeats**

**Display column heading**

**FOR the total rows**

**Display row heading**

**FOR the total columns**

**Set spacing between A’s within field**

**ShowSeat::GetSeat**

**Prompt for and accept number of tickets**

**FOR the number of tickets**

**Output current ticket customers selecting seat for**

**Prompt for and accept desired seat row number and column number**

**FOR check along row**

**FOR check along column**

**IF user seat input matches field currently scanning**

**Replace A with a H**

**Return number of tickets variable**

# Ticket.h

**Ticket::SetPrice**

**Output customers number of tickets**

**Prompt for and accept quantity of child, student, adult and senior tickets**

**IF customer number of tickets matches quantity of c, s, a, & s tickets**

**Calculate total price**

**Output price to customer**

**ELSE IF number of tickets is too high in comparison**

**Output too many tickets**

**Call set price function**

**ELSE IF number of tickets is too low in comparison**

**Output not selected enough tickets**

**Call set price function**

**ELSE**

**Output invalid input**

**Call set price function**

**Ticket::Deselect**

**Prompt for and accept whether customer still wants tickets**

**SWITCH (Integer variable)**

**CASE 1: Output customers decision to buy tickets**

**CASE 2: Output customers decision to not buy and exit console**

**DEFAULT: Output ‘Invalid input’, call to Deselect function**

**Ticket::Pay**

**Prompt for and accept card details**

**Ticket::PrintTicket**

**Display ticket**