Bryan Wong

805111517

Professor Stahl

CS31

February 27, 2019

Project 5 Report

a. During this project, I ran into a few notable obstacles. Most of these struggles came from dealing with the multiple files and ensuring that no linker errors occurred. Over the process of coding project 5, I encountered these errors from mistakes such as using #include on .cpp files and their respective .h files, or not being careful with variable and object scopes. Working with the newly introduced enum values was also a challenge. It took experimentation to realize that functions returning enumerated variables (for example, to return a KIND) had to be in the following format in the .cpp implementation:

Ticket::KIND Ticket::getKind( )

Other pitfalls included learning all the syntax for .h files, for example class Foo { }; , public: and private:, #ifndef and #endif, etc. The actual implementation went quite smoothly, once I understood the structure of the BoxOffice functions. For example, buyRoyceHallTicket( ) can use buyTicket( ) and modify its price to achieve the desired result.

b. Here is a list of several test cases I used to test my code:

**Test that all ticket functions (getters and setters) work correctly:**

Ticket test("Orchestra Row B", 32, "Derrick's Birthday Party", "The Rat Den", 959.99, Ticket::OTHER, "December 1, 2:30AM");

assert(test.getRow() == "Orchestra Row B");

assert(test.getSeat() == 32);

assert(test.getEvent() == "Derrick's Birthday Party");

assert(test.getVenue() == "The Rat Den");

assert(test.getPrice() == 959.99);

assert(test.getKind() == Ticket::OTHER);

assert(test.getDayTime() == "December 1, 2:30AM");

test.setPrice(50.00);

assert(test.getPrice() == 50.00);

**This test case was given in the spec. It is once again used to test that the ticket functions work properly:**

Ticket t( "Pavillion Row 13", 45, "Dodgers vs. Brewers", "Dodger Stadium", 150.00, Ticket::ATHLETICGAME, "August 1, 7:30PM" );

assert( t.getPrice() == 150.000000 );

assert( std::to\_string( t.getPrice() ) == "150.000000" );

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Pavillion Row 13" );

assert( t.getSeat() == 45 );

assert( t.getEvent() == "Dodgers vs. Brewers");

assert( t.getVenue() == "Dodger Stadium");

assert( t.getDayTime() == "August 1, 7:30PM");

**The following test cases rigorously test the BoxOffice functions, making sure that each parameter is correctly assigned and that the price of the ticket is calculated correctly. For each KIND, AM/PM, and ORCHESTRA/BALCONY seating, the price that the ticket is set to is checked:**

**// Non-Royce Hall Ticket ( uses buyTicket( ) ) - Price defaults to 0**

t = b.buyTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", "The Rat Den", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "0.000000" );

assert( t.getPrice() == 0 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "The Rat Den");

assert( t.getDayTime() == "September 1, 7:30PM");

**// ATHLETIC - PM - ORCH: Given these parameters (the only ones that matter in setting prices), we check that the price of the ticket is accurate, and that the parameters carry over. This is repeated for every combination of these factors.**

t = b.buyRoyceHallTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "150.000000" );

assert( t.getPrice() == 150);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "120.000000" );

assert( t.getPrice() == 120);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "130.000000" );

assert( t.getPrice() == 130);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

**// ATHLETIC - PM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "75.000000" );

assert( t.getPrice() == 75);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "45.000000" );

assert( t.getPrice() == 45);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "55.000000" );

assert( t.getPrice() == 55);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

**// ATHLETIC - AM - ORCH : AM does not affect price for ATHLETICGAME**

t = b.buyRoyceHallTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "150.000000" );

assert( t.getPrice() == 150);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "120.000000" );

assert( t.getPrice() == 120);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "130.000000" );

assert( t.getPrice() == 130);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

**// ATHLETIC - AM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "75.000000" );

assert( t.getPrice() == 75);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "45.000000" );

assert( t.getPrice() == 45);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Balcony Row A", 12, "World Cup", Ticket::ATHLETICGAME, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "55.000000" );

assert( t.getPrice() == 55);

assert( t.getKind() == Ticket::KIND::ATHLETICGAME );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "World Cup");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

**// CONCERT - PM - ORCH**

t = b.buyRoyceHallTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "130.000000" );

assert( t.getPrice() == 130);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "100.000000" );

assert( t.getPrice() == 100);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "110.000000" );

assert( t.getPrice() == 110);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

**// CONCERT - PM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "65.000000" );

assert( t.getPrice() == 65);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "35.000000" );

assert( t.getPrice() == 35);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "45.000000" );

assert( t.getPrice() == 45);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30PM");

**// CONCERT - AM - ORCH**

t = b.buyRoyceHallTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "100.000000" );

assert( t.getPrice() == 100);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "70.000000" );

assert( t.getPrice() == 70);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "80.000000" );

assert( t.getPrice() == 80);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Orchestra Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

**// CONCERT - AM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "50.000000" );

assert( t.getPrice() == 50);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "20.000000" );

assert( t.getPrice() == 20);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Balcony Row A", 12, "Barry Manilow Live In Concert", Ticket::CONCERT, "November 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "30.000000" );

assert( t.getPrice() == 30);

assert( t.getKind() == Ticket::KIND::CONCERT );

assert( t.getRow() == "Balcony Row A" );

assert( t.getSeat() == 12 );

assert( t.getEvent() == "Barry Manilow Live In Concert");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "November 1, 7:30AM");

**// MOVIE - PM - ORCH**

t = b.buyRoyceHallTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "25.000000" );

assert( t.getPrice() == 25.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "20.000000" );

assert( t.getPrice() == 20.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "22.500000" );

assert( t.getPrice() == 22.500000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

**// MOVIE - PM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "12.500000" );

assert( t.getPrice() == 12.500000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "10.000000" );

assert( t.getPrice() == 10.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "11.250000" );

assert( t.getPrice() == 11.250000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30PM");

**// MOVIE - AM - ORCH : AM should not change the price**

t = b.buyRoyceHallTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "25.000000" );

assert( t.getPrice() == 25.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "20.000000" );

assert( t.getPrice() == 20.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "22.500000" );

assert( t.getPrice() == 22.500000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Orchestra Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

**// MOVIE - AM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "12.500000" );

assert( t.getPrice() == 12.500000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "10.000000" );

assert( t.getPrice() == 10.000000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Balcony Row U", 20, "Episode VII: The Force Awakens", Ticket::MOVIE, "September 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "11.250000" );

assert( t.getPrice() == 11.250000 );

assert( t.getKind() == Ticket::KIND::MOVIE );

assert( t.getRow() == "Balcony Row U" );

assert( t.getSeat() == 20 );

assert( t.getEvent() == "Episode VII: The Force Awakens");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "September 1, 7:30AM");

**// OTHER - PM - ORCH**

t = b.buyRoyceHallTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "110.000000" );

assert( t.getPrice() == 110);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "80.000000" );

assert( t.getPrice() == 80);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "90.000000" );

assert( t.getPrice() == 90);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

**// OTHER - PM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "55.000000" );

assert( t.getPrice() == 55);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

t = b.buyRoyceHallStudentTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "25.000000" );

assert( t.getPrice() == 25);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

t = b.buyRoyceHallStaffTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30PM" );

assert( std::to\_string( t.getPrice() ) == "35.000000" );

assert( t.getPrice() == 35);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30PM");

**// OTHER - AM - ORCH**

t = b.buyRoyceHallTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "80.000000" );

assert( t.getPrice() == 80);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "50.000000" );

assert( t.getPrice() == 50);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Orchestra Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "60.000000" );

assert( t.getPrice() == 60);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Orchestra Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

**// OTHER - AM - BALCONY**

t = b.buyRoyceHallTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "40.000000" );

assert( t.getPrice() == 40);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

t = b.buyRoyceHallStudentTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "10.000000" );

assert( t.getPrice() == 10);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

t = b.buyRoyceHallStaffTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "20.000000" );

assert( t.getPrice() == 20);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");

**// ODDS AND ENDS: Other edge cases are tested here**

**// Check empty strings and negative price/seat values**

Ticket test2( "", -1, "", "", -1, Ticket::ATHLETICGAME, "" );

assert( test2.getPrice() == -1 );

assert( std::to\_string( test2.getPrice() ) == "-1.000000" );

assert( test2.getKind() == Ticket::KIND::ATHLETICGAME );

assert( test2.getRow() == "" );

assert( test2.getSeat() == -1 );

assert( test2.getEvent() == "");

assert( test2.getVenue() == "");

assert( test2.getDayTime() == "");

**// Check for seat and price with values 0**

Ticket test3( "", 0, "", "", 0, Ticket::ATHLETICGAME, "" );

assert( test3.getPrice() == 0 );

assert( std::to\_string( test3.getPrice() ) == "0.000000" );

assert( test3.getKind() == Ticket::KIND::ATHLETICGAME );

assert( test3.getRow() == "" );

assert( test3.getSeat() == 0 );

assert( test3.getEvent() == "");

assert( test3.getVenue() == "");

assert( test3.getDayTime() == "");

**// If AM/PM is not specified, default to AM pricing**

t = b.buyRoyceHallTicket( "Balcony Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30" );

assert( std::to\_string( t.getPrice() ) == "40.000000" );

assert( t.getPrice() == 40);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Balcony Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30");

**// If Orchestra/Balcony is not specified, defaults to Balcony pricing**

t = b.buyRoyceHallTicket( "Row J", 120, "An Evening With David Sedaris", Ticket::OTHER, "October 1, 7:30AM" );

assert( std::to\_string( t.getPrice() ) == "40.000000" );

assert( t.getPrice() == 40);

assert( t.getKind() == Ticket::KIND::OTHER );

assert( t.getRow() == "Row J" );

assert( t.getSeat() == 120 );

assert( t.getEvent() == "An Evening With David Sedaris");

assert( t.getVenue() == "Royce Hall");

assert( t.getDayTime() == "October 1, 7:30AM");