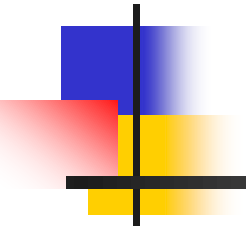
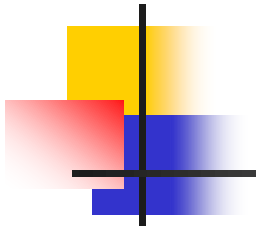


Project 3: Ticket Printer

Project 3: Ticket Printer



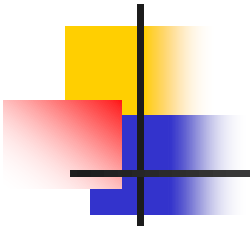
- Write a program to produce a set of tickets for a specific performance at a specific venue.

- You can modify the Ticket class from Project 1 for use in this project.
  - OK to use posted solution.

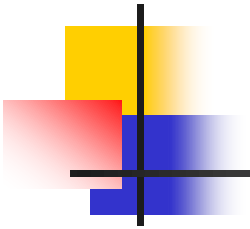
Class Venue

- Write a definition for class Venue.
- A Venue object corresponds to a physical site for a performance.
- Attributes of a Venue are:
  - Name
  - Address (Described later)
  - A collection of Seats
    - Details on next slide
  - Capacity (total number of seats)
    - Capacity is computed from the seats collection.

Class Venue

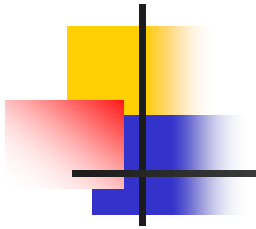


- Let a Venue have a collection of Seat_Row objects.
  - Maximum of 1000 rows.
- Each Seat_Row object has
  - A row name
  - Number of seats.
  - A collection of seats.
    - Seats in each row are numbered consecutively starting with 1.
    - Maximum of 1000 seats in a row.



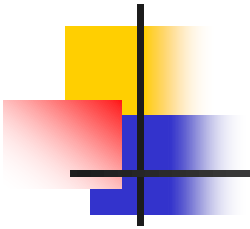
Class Address

- Define a class to represent Addresses
- Attributes of an Address
  - Street address
  - City
  - State – 2 characters
  - Zip code – 5 digits
- Street address, City, and State should be C++ strings.
- Provide accessor functions to get (but not set) the values of all attributes.



Class Venue

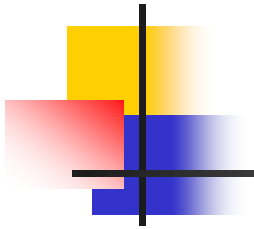
- Design diagram for Venue
  - Classroom discussion.



Class Performance

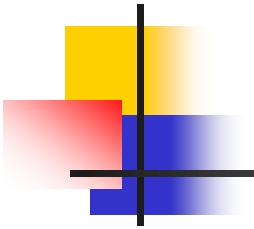
- Define a class to represent performances

- Attributes of a Performance
  - Show Name (C++ string)
  - Venue
  - Date – Day, Month, Year (integers)
  - Time – Hour, Minute (integers)
    - Date and Time are the same as in Project 1



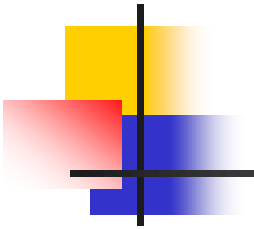
Class Performance

- Design diagram for Performance
  - Classroom discussion.



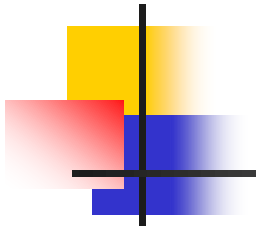
Class Ticket

- Attributes of a Ticket:
  - Performance
    - Includes Venue and date/time
  - Seat
  - Sold (boolean)



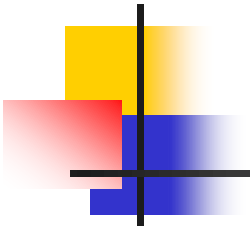
Class Ticket_Book

- A Ticket_Book object holds tickets for all seats of a specific Performance.
- Display method outputs a complete set of tickets.
  - For this project, output to the screen.



Class Ticket_Book

- Design diagram for Ticket_Book
  - Classroom discussion.



Program Ticket_Printer

- Your main() function should produce a Ticket_Book for a performance of “Billy Elliot” at The Little Theater on April 2, 2016 at 8:00 PM.
- The Little Theater is located at 19 Foster Street, Littleton, MA, 01460.
- The Little Theater has 3 rows, with names A through C.
- Each row has 4 seats.

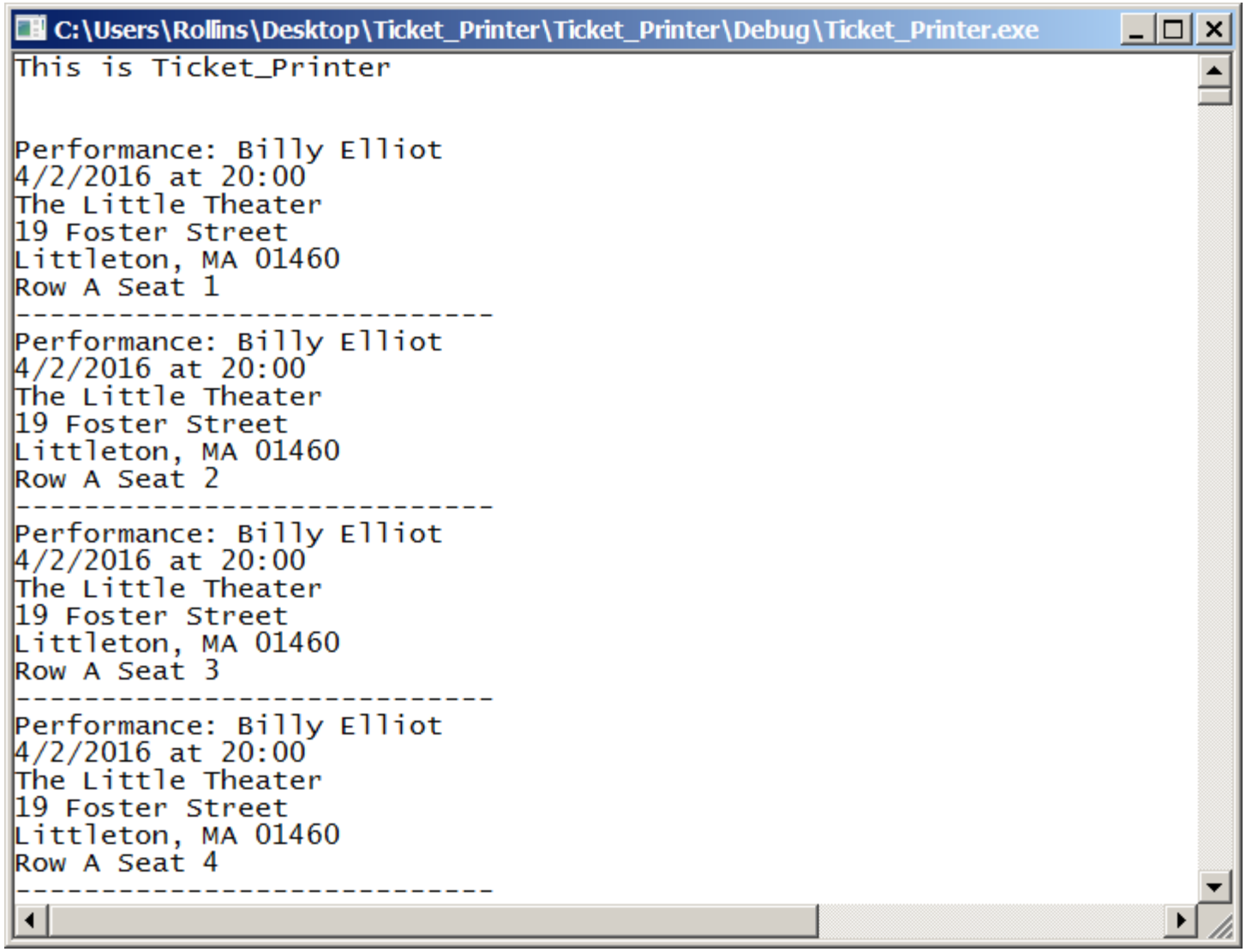
Program Ticket_Printer

- Use the main() function on the following slide.
  - Create a Venue object.
  - Create a Performance object.
  - Create a Ticket_Book object.
  - Use the Ticket_Book object to display the tickets on the screen.
- There is no user input.
  - All information must be built in.
  - A more realistic program would get the needed information from a file.

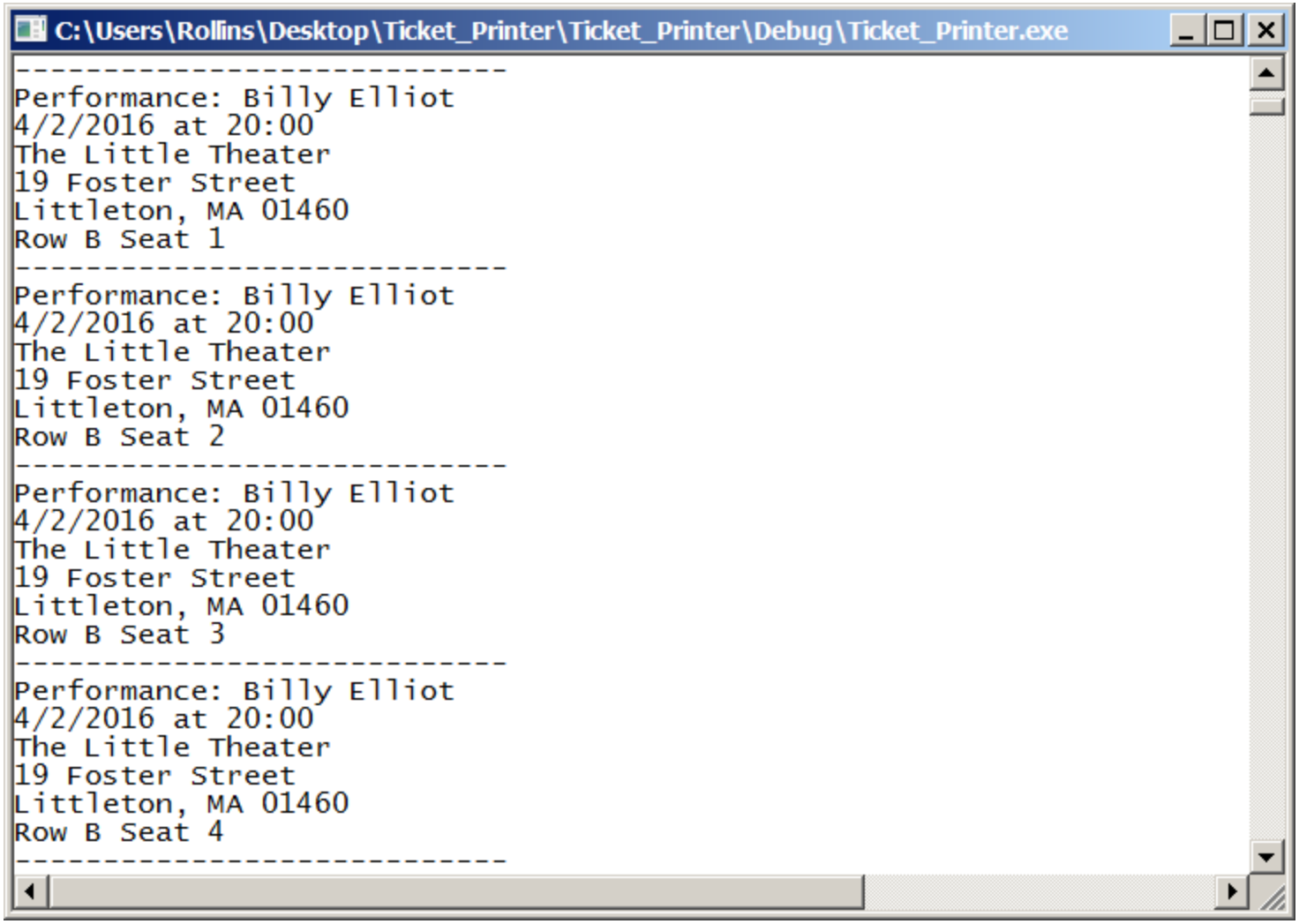
main()

```
int main()
{
    cout << "This is program Ticket_Printer\n\n\n";

    Venue* venue = Create_Venue();

    Performance* performance = Create_Performance(venue);

    Ticket_Book* ticket_book = new Ticket_Book(performance);

    ticket_book->Display();

    cin.get();
    return 0;
}
```

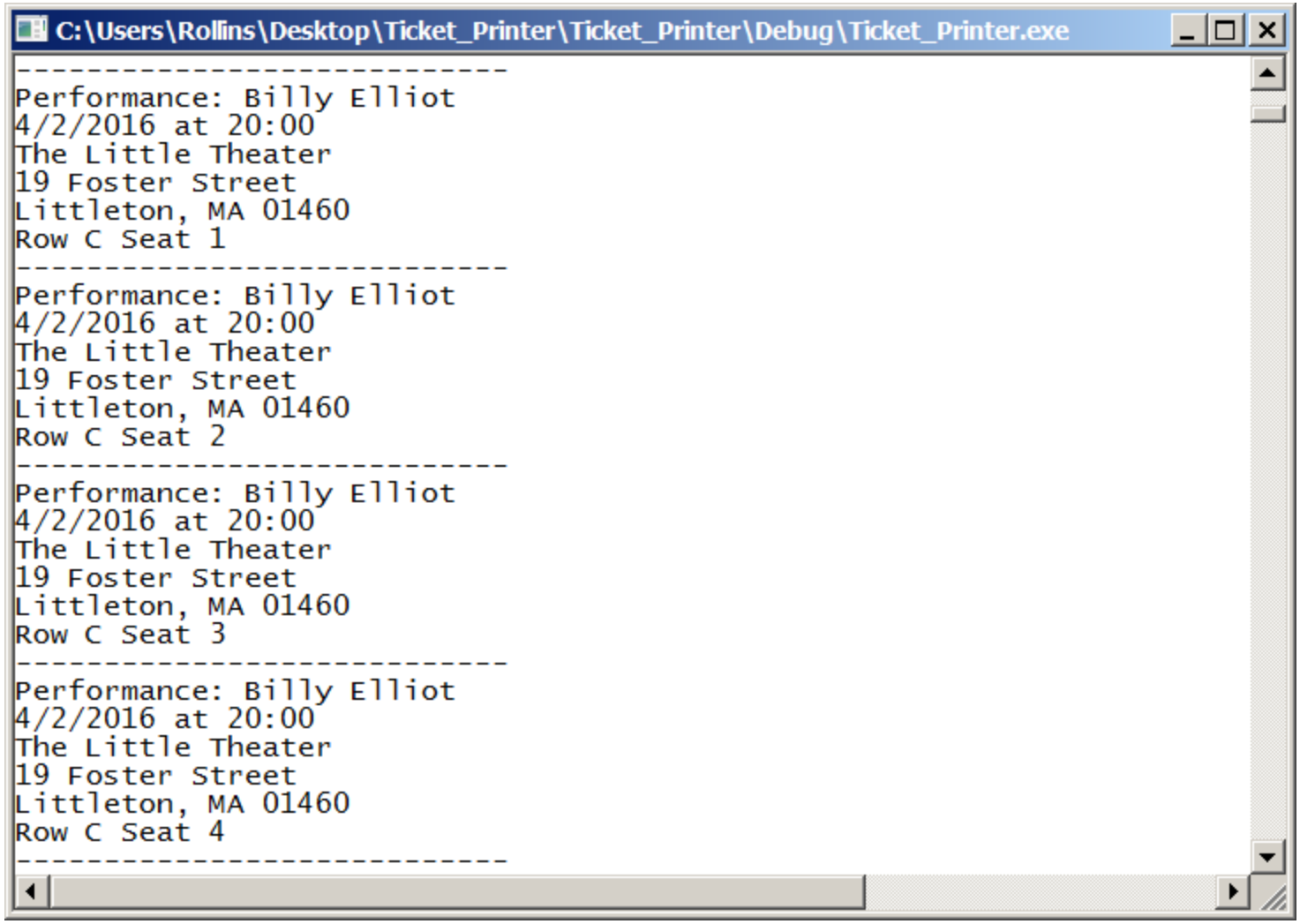
Program Output

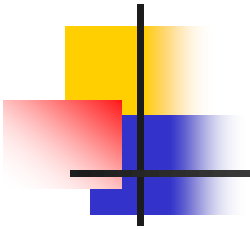


Program Output



Program Output





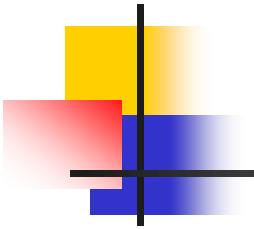
Development Environment

- You may develop your program on any system you like.
- But you should test the finished program on Circe.

- The same source files should compile and run on *either Windows or Linux*.

Ground Rules

- You may work with one other person.
  - OK to work alone if you prefer.
- If you do work as a pair
  - Both members are expected to contribute.
  - Submit a single program.
  - Both members should understand the program in detail.
- Do not share your code with other students.
  - Before or after submitting the project.
  - OK to *discuss* the project.
- Do not copy any other student's work.
  - Don't *look at* anyone else's program.
  - Don't let anyone look at your program.



Ground Rules

Except for code posted on the class web site

- Do not copy code from the Internet
  - or any other source.

- Write your own code.

Submission

- Project is due by 11:59 PM, Thursday, Feb. 11.
- Deliverables:
  - Source code only.
  - Zip the files for submission.
    - Please put your source files into a folder
    - Use the Windows "Send to Compressed Folder" command
    - Do not submit any other form of zipped folder
    - If you have trouble zipping the files, submit the separate files.
- If you work with another student, include both names in the Canvas submission comments.
  - Other student should submit just a Blackboard comment including both names (if possible)

End of Presentation