Bingo Game Instructions

Write implementation for every method in the assignment project that contains the comment tag STUB.
Keep track of your progress using version control.
Run code coverage to ensure the code you have written is tested.
Use various software engineering tools to help create quality software (static and style analysis, memory leak checking,
continuous integration

Check that the method you have written.
(c1) Passes the staticAnalysis test (make static).
(c2) Passes the styleCheck (make style).
(c3) Compiles using the Makefile (make test_bingo).
(c4) Passes the unitTests (./test_bingo) for that method.
(c5) Has near 100% line coverage (make coverage).
(c6) Has no memory leaks (make memcheck).

Implement code in the following order:
(d1) Constructors
(d2) Mutator/setter methods used by the constructor, if any.
(d3) Destructors
(d4) Accessor/getter methods
(d5) Mutator/setter methods
(d6) Methods that do not depend on incomplete methods.

Files

//header files

BingoCaller.h
BingoCard.h
BingoCardFactory.h
BingoGame.h
BingoTypes.h
DaubState.h
Exceptions.h
MakeRandomInt.h
ScreenDisplay.hg
Square.h

UserInput.h
VictoryCondition.h

.cpp files

BingoCaller.cpp
BingoCard.cpp
BingoCardFactory.cpp
BingoGame.cpp
BingoTypes.cpp
DaubState.cpp
MakeRandomInt.cpp
ScreenDisplay.cpp
Square.cpp
UserInput.cpp
VictoryCondition.cpp

Expected outcomes :

BingoCardtest.expected

```
+------+------+------+------+------+
|   B  |   I  |   N  |   G  |   O  |
+------+------+------+------+------+
|  04  |  22  |  45  |  46  |  65  |
|  06  |  27  | (43) |  54  |  67  |
|  05  |  19  | free | (49) |  70  |
|  07  | (20) |  38  |  56  |  74  |
|  02  |  16  |  41  | (58) | (73) |
+------+------+------+------+------+
```

bingoInstruction.expeced

```
On your turn, the caller will announce the ball chosen.
 - Your card will be displayed.
 - You will be asked which player action you wish to take.
 - Try to daub only the numbers called.
 - If a number on your card has previously been daubed it will appear in
braces.
 - The goal is to be the first to daub the desired pattern for this game.
 - If you believe you have won choose the Bingo action.
 - Your card will be checked to see if you match the victory condition.
```

```
 - If your card is correct then the game is over, otherwise play continues.
```

ValidityDisplayTest.expected

```
+------+------+------+------+------+
|   B  |   I  |   N  |   G  |   O  |
+------+------+------+------+------+
|  04  |  22  |  45  |  46  |  65 x|
|  06  |  27  | (43) |  54  |  67  |
|  05  |  19  | free |  49  |  70  |
|  07  |  20  |  38  |  56  |  74  |
|  02  |  16  |  41  |  58  | (73)x|
+------+------+------+------+------+
```

Data

BingoInfo.txt

```
TITLE 3
==========================
===== Bingo Game Demo =====
==========================
SETUP 5
1. Choose a game type
  - Bingo50 uses 50 balls labelled with letters & arithmetic expressions.
  - Bingo75 uses 75 balls labelled with letters & numbers.
2. Add players until everyone has a card.
3. Start the game.
PLAY 9
On your turn, the caller will announce the ball chosen.
 - Your card will be displayed.
 - You will be asked which player action you wish to take.
 - Try to daub only the numbers called.
 - If a number on your card has previously been daubed it will appear in
braces.
 - The goal is to be the first to daub the desired pattern for this game.
 - If you believe you have won choose the Bingo action.
 - Your card will be checked to see if you match the victory condition.
 - If your card is correct then the game is over, otherwise play continues.
```

Bingo Game Project

Specifications

Sample output

- This example shows one bingo card.
- A horizontal line made of plus signs and dashes appears at the top, bottom, and after the header.
- The header displays the column letters B, I, N, G, and O.
- Daubed squares are indicated by encasing the number in round braces.
- The free square is always daubed.
- Vertical bars are used as column delimiters.
- The interior of each column has a width of 6 characters.
    - The first and last character are always blank.
    - The numbers are all displayed in 2 digits.
    - All numbers and letters are right aligned at the fourth character of the column.

```
+------+------+------+------+------+
|   B  |   I  |   N  |   G  |   O  |
+------+------+------+------+------+
|  04  |  22  |  45  |  46  |  65  |
|  06  |  27  | (43) |  54  |  67  |
|  05  |  19  | free | (49) |  70  |
|  07  | (20) |  38  |  56  |  74  |
|  02  |  16  |  41  | (58) | (73) |
+------+------+------+------+------+
```