

STUDENT

Qingxiao Dong

COURSE

Intro to Java Programming

---

Hello Qingxiao,

Thanks for making those improvements to your final project, and I'm happy to say that your submission passes specifications! You've done a great job creating an agent that dominates at a game of Connect Four all with readable Java code and documentation.

Again, congratulations! And do keep us updated on your future programming endeavors and stay Udacious.

Cheers,

Jessica

---

**Code Functionality****Exceeds Specifications**

- The code compiles and runs without throwing any Exceptions or Errors. These include but are not limited to NullPointerExceptions, Compile errors, Runtime errors, etc.
- The agent has a functional move method.
- The agent recognizes and responds appropriately to available moves that would cause either player to win the game.
- The agent can reliably win the game. The agent beats the Random Agent at least 18 times out of 20 and the Beginner Agent at least 12 times out of 20. Also, the agent beats one of the more difficult opponent agents (e.g. Brilliant Agent) at least 12 times out of 20.
  - Your Agent has amazing winning stats! Out of 1000 games against the Beginner Agent, yours won 98.4% of the time. Out of 1000 games against the Random Agent, yours won 100% of the time, out of 1000 games against the Intermediate Agent, yours won 97.6% of the time, and out of 1000 games against the Brilliant Agent, yours won 94.4% of the time. Good work on your game logic.

### **Use of Control Flow Statements**

### **Meets Specifications**

- Appropriate control statements (for, while, if, else) are used in each relevant situation.
- Iterative control statements (for, while) are used effectively to avoid repetitive code.
- Conditional control statements (if, else) effectively used to guide code flow.

### **Definition and Use of Methods**

### **Meets Specifications**

- Repeated blocks of code are encapsulated in methods.
- Methods defined in the program are called where appropriate.
- Parameters and return values are appropriate for each method's purpose.
- Use static and instance members of a class properly.

### **Object Oriented Programming**

### **Meets Specifications**

- Code correctly uses accessor methods to get game data out of the board.
- Code correctly uses mutators to modify the game board.
- Code efficiently and effectively leverages the object-oriented class structure of the game.

### **Code Readability**

### **Meets Specifications**

- Naming conventions from the Java Language Coding Guidelines are used:
  - Variable and method names are lowercase, with occasional upperCase characters in the middle.
  - Class names start with an Uppercase letter.
  - Constant names are UPPERCASE, with an occasional UNDER\_SCORE, and are declared as final.
- Method documentation is clear and concise and is included for each method.
- Names of methods and parameters are descriptive and meaningful.
- Magic numbers are avoided. final variables are defined for numeric constants used in the program.

## PROJECT EVALUATION

### **Project Meets Specifications**

