CSE 114, Computer Science 1
Course Information

Fall 2017

Stony Brook University

Instructor: Dr. Praveen Tripathi

http://www.cs.stonybrook.edu/~cse114

Course Description

- "Procedural and object-oriented programming methodology. Topics include program structure, conditional and iterative programming, procedures, arrays and records, object classes, encapsulation, information hiding, inheritance, polymorphism, file I/O, and exceptions. Software debugging and testing techniques are emphasized. ***Includes required laboratory.***" (https://www.cs.stonybrook.edu/students/Undergraduate-Studies/courses/CSE114)
- Prerequisites: Level 4 or higher on the math placement exam (or MAT 123+)
- **Advisory Prerequisite: CSE 101 or ISE108.**

Course Outcomes

- The following are the official course goals agreed upon by the faculty for this course:
  - An ability to program in an object oriented language, using concepts such as object classes, encapsulation, inheritance, and polymorphism.
  - An ability to use fundamental data structures such as arrays.
  - An ability to program with sound code structure and use systematic software debugging and testing techniques.

Course Focus

- Introduction to programming (in Java):
  - conditional statements
  - loops
  - methods
- Fundamental data structures of high-level programming: arrays, lists, stacks
- Algorithms
- Basic concepts of object-oriented programming
  - object classes
  - encapsulation
  - inheritance
  - polymorphism
  - Application: GUIs
- Programming assignments
  - systematic software debugging
  - systematic testing techniques

Major Course topics

1. **Procedural Programming Basics:**
   - **data types**
   - **variable declarations**
   - **assignment statements & expressions**
   - **input/output**
   - **textual manipulation & strings**
   - **conditional (branching) statements**
   - **iteration = loops and recursion**
   - **method construction**

Major Course topics

2. **Arrays:**
   - **collect data in arrays**
   - **searching**
   - **sorting**
   - **array manipulations**

Major Course topics

3. **Object Oriented Programming:**
   - **designing and constructing classes**
   - **aggregation**
   - **inheritance**
   - **polymorphism**
   - **Application: GUIs**

Instructor Information

- Dr. Praveen Tripathi
  202 New Computer Science Building
- Office hours: Tuesdays and Thursdays 2:30PM-4:00PM
- Phone: 1 (704) 858-7047
- Email: praveen(at)cs(dot) stonybrook (dot) edu
  - Please include "CSE 114" in the email subject and your name in your email correspondence
- TAs: see course Web page:
  http://www.cs.stonybrook.edu/~cse114

General Information

- Course Web page: http://www.cs.stonybrook.edu/~cse114
  - **Blackboard** will be used for assignments, grades and course material.

General Information

- Meeting Information (Class Time and Place):
  - CSE 114-01 (89940) Computer Science I (Lecture 1): MoWe7:00PM - 8:20PM, Engineering 143.
  - CSE 114-L01 (89941) Computer Science I (Laboratory): MoWe 8:30AM - 9:50AM, Computer Science building, room 2116.
  - CSE 114-L02 (89942) Computer Science I (Laboratory): MoWe 10:00AM - 11:20AM, Computer Science building, room 2116.
  - CSE 114-L03 (89943) Computer Science I (Laboratory): MoWe 11:30AM - 12:50PM, Computer Science building, room 2116.
  - CSE 114-L04 (89944) Computer Science I (Laboratory): MoWe 2:30PM - 3:50PM, Computer Science building, room 2116.
  - CSE 114-L05 (89945) Computer Science I (Laboratory): MoWe 4:00PM - 5:20PM, Computer Science building, room 2116.
  - CSE 114-L06 (91546) Computer Science I (Laboratory): MoWe 5:30PM - 6:50PM, Computer Science building, room 2116.
    - **Computer Science 2116 is the Computer Science SINC site.**

Textbook

- Intro.To Java Programming, Brief Vers. , Author: Liang , Publisher: Pearson , Edition: 10th, 2014.
  - ISBN 9780133813470 is the Student Value Edition for Introduction to Java Programming, Brief Version plus http://www.MyProgrammingLab.com
  - MyProgrammingLab.com is the online testing system that comes with the textbook and we will use it for online computer programming exercises executed in the laboratory and at home as part of the homework assignments.
  - The book with ISBN 9780133813470 includes the subscription to MyProgrammingLab.
  - Students who wish to purchase accss to MyProgrammingLab without the textbook may do so by visiting theMyProgrammingLab.com website provided by the Pearson publisher of our textbook.

Software

- Necessary Software:
  - Java Developer Kit (JDK): download from http://java.com/en/download/index.jsp
    - You should download JDK for your operating system (cost: free)
  - Eclipse IDE: http://www.eclipse.org
    - You should download the Eclipse IDE for Java Developers (cost: free)

Coursework

- Grading Schema:
  - Homework, project, quizzes and labs=25%
    - Programming homework assignments
    - Project
    - Class quizzes
    - Lab assignments
  - Midterm exams (2) = 50% (25% each)
  - Final exam = 25%

Important Dates

- Midterm Exam #1: Evening exam, Monday, 10/16, from 8:45 pm - 10:05 pm (80 minutes)
- Midterm Exam #2: Evening exam, Tuesday, 11/7, from 8:45 pm - 10:05 pm (80 minutes)
- Final Exam: Common exam, Wednesday, December , 2017, 8AM-10AM (2 hours = 120 min final exam)
  - See SOLAR for evening and common exams schedules: https://it.stonybrook.edu/services/solar
  - See Final Exams University Schedule for final exam schedules: http://www.stonybrook.edu/registrar/finals.shtml
  - The exams will be like the problems that we solve in the class!

Assignments

- Homework assignments due on fixed dates and times.
  - **no late submission is permitted**
- All assignments should be submitted electronically
  - Blackboard and the textbook Web site

Lab exercises

- Simple Coding Exercises done in Computer Science (CS) SINC-site room CS 2116
  - You have only the lab-hour to edit, compile and execute your solution
  - Attendance is mandatory, if you want credit
    - you can leave early only if you showed all your assigned work
    - if you come late, then you don't get the credit for the lab
      - you can come to another lab that same day to get the credit for the lab () same rules apply as above!
  - Demonstrate your work to Lab-TA before you leave for lab credit

    0 –3 points:
    - 0 - Student did not attend the lab or program does not even compile.
    - 1 - Student attended the lab, program compiles but has major problems.
    - 2 - Student attended the lab, and program partially works (with some minor errors)
    - 3 - Student attended the lab, and program is correct

Regrading of Homework/Exams

- Please meet with a grading TA or the instructor and arrange for regrading.
- **You have one week from the day grades are posted or mailed or announced!**
  - Late requests will not be entertained

Class Schedule

Week	Lecture Topics
1	Introduction to Computers, Programming and Java
2	Elementary Programming and Selections
3	Mathematical Functions, Characters, and Strings, Loops
4	Methods
5	Arrays and Multi-dimensional Arrays
6	Arrays and Multi-dimensional Arrays
7	Objects and Classes, Object-Oriented Thinking
8	Objects and Classes, Object-Oriented Thinking
9	Inheritance and Polymorphism
10	Exception Handling and Text I/O
11	Abstract Classes and Interfaces
12	JavaFX Basics, Event-Driven Programming
13	JavaFX UI Controls and Multimedia
14	Recursion
15	Recursion

Disability Support Services

- If you have a physical, psychological, medical or learning disability, contact the DSS office at Room 128 ECC. Phone 632-6748/TDD
- If you are planning to take an exam at DSS office, you need to tell me ahead of time for every exam.
- **All documentation of disability is confidential.**

Academic Integrity

- You can discuss general assignment concepts with other students: explaining how to use systems or tools and helping others with high-level design issues
- You **MAY NOT share** assignments, source code or other answers by copying, retyping, looking at, or supplying a file
  - Assignments are subject to manual and automated similarity checking (We do check! and our tools for doing this are much better than cheaters think)
- If you cheat, you will be brought up on academic dishonesty charges - we follow the university policy:
  - http://www.stonybrook.edu/uaa/academicjudiciary

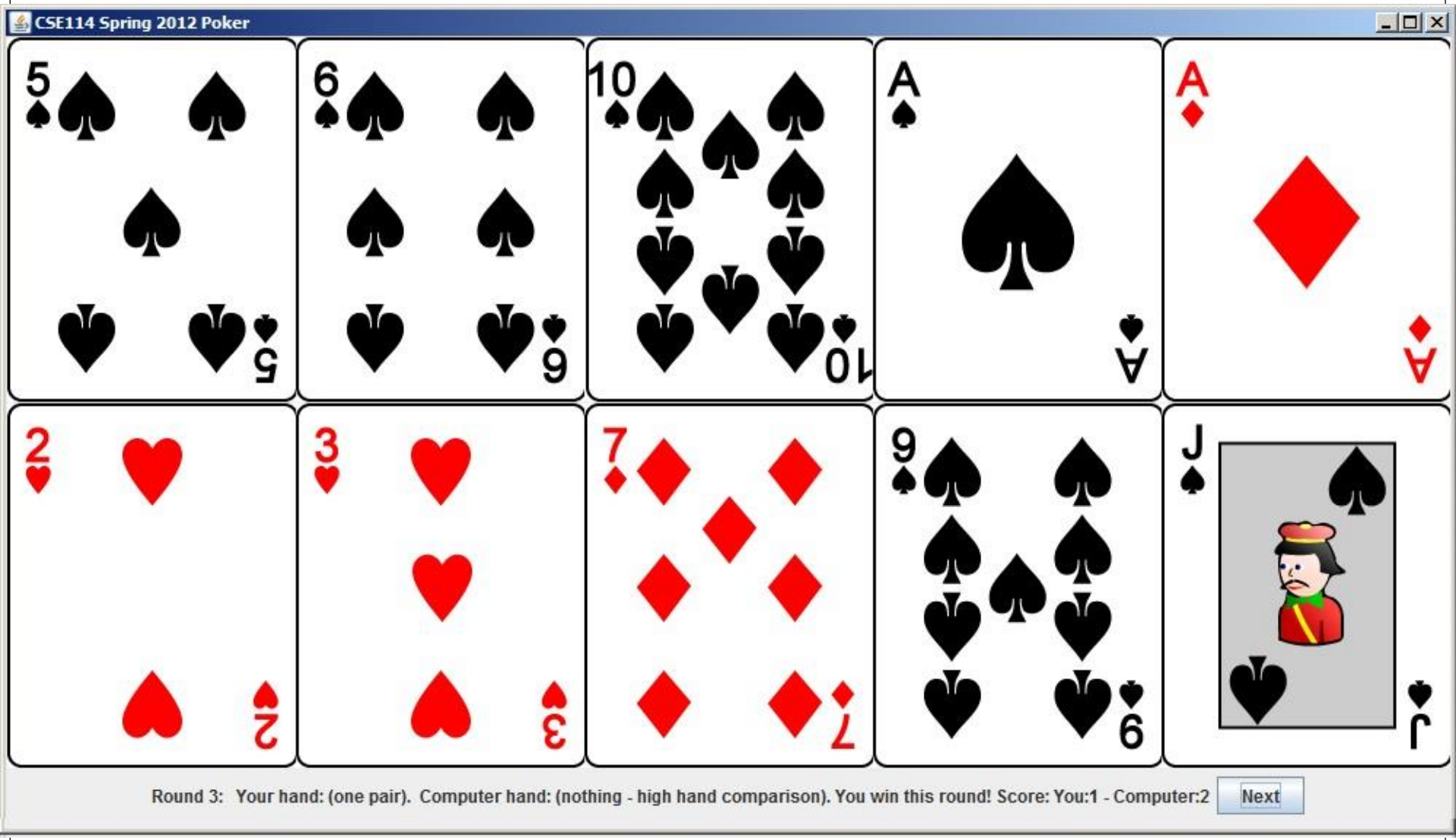
Catastrophic events

- Major illness, death in family
- Formulate a plan (with your CEAS academic advisor) to get back on track
- Advice
  - Once you start running late, it's really hard to catch up

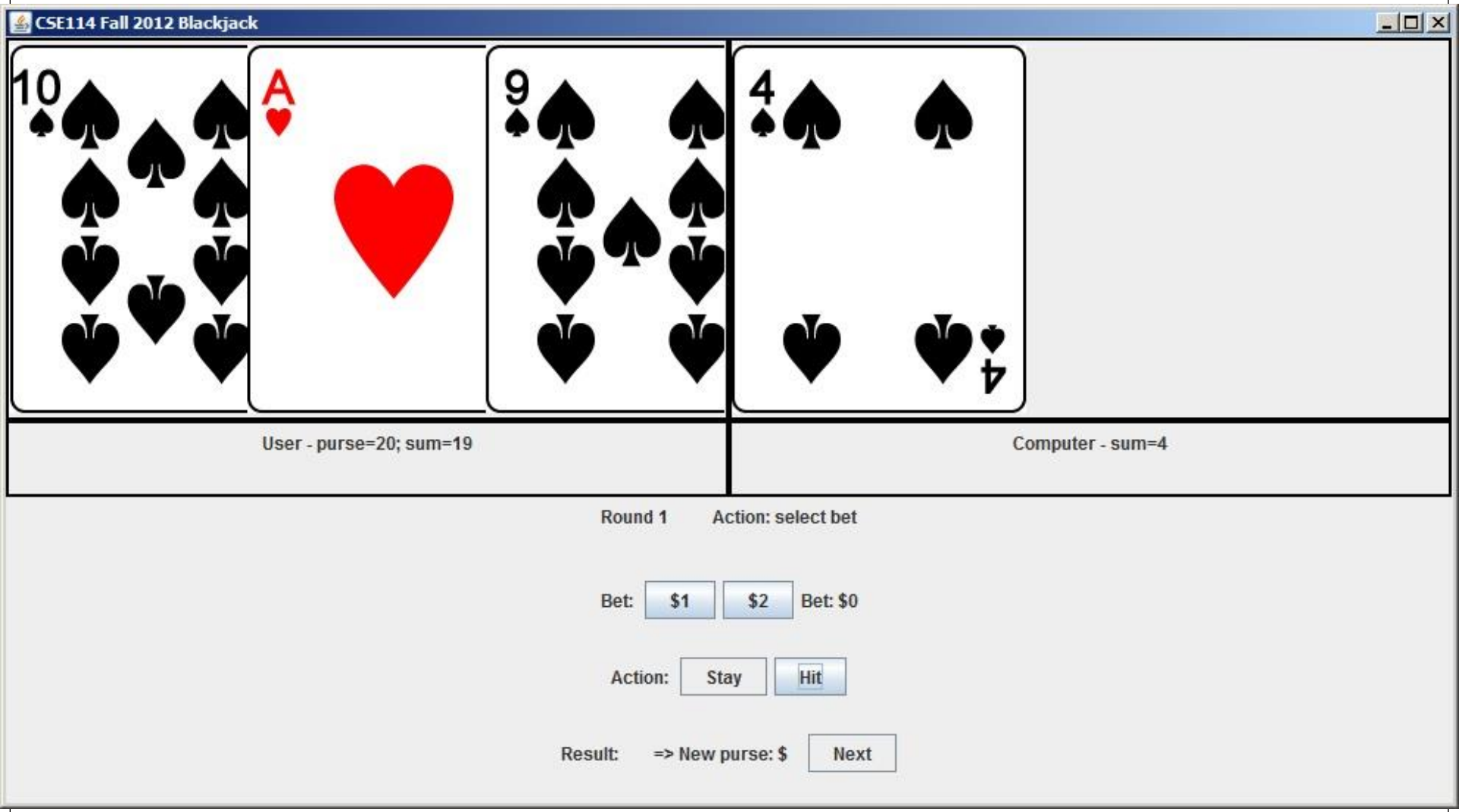
What do you need to get started?

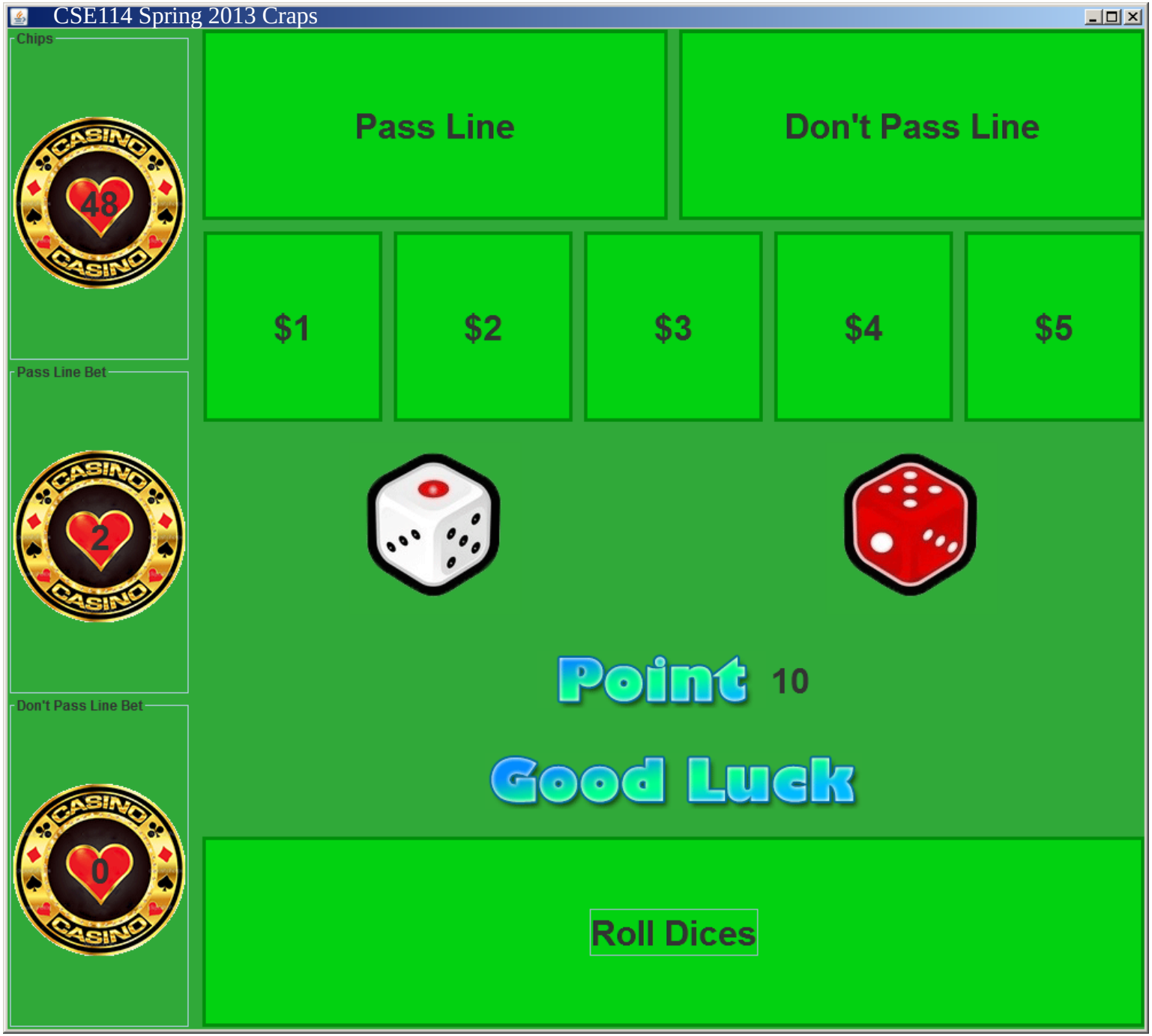
- Blackboard account
  - http://blackboard.stonybrook.edu
- SINC Sites: http://www.sinc.sunysb.edu/helpdesk/labs.shtml
- Java JDK standard edition:
  - http://www.oracle.com/technetwork/java/javase/downloads
- Eclipse IDE:
  - http://www.eclipse.org/downloads
  - Learn to use the debugger!!!
- Liang's student Web site:
  - http://www.cs.armstrong.edu/liang/intro10e

Past CSE114 Projects: Spring 2012: Artificial Intelligence (AI) Poker

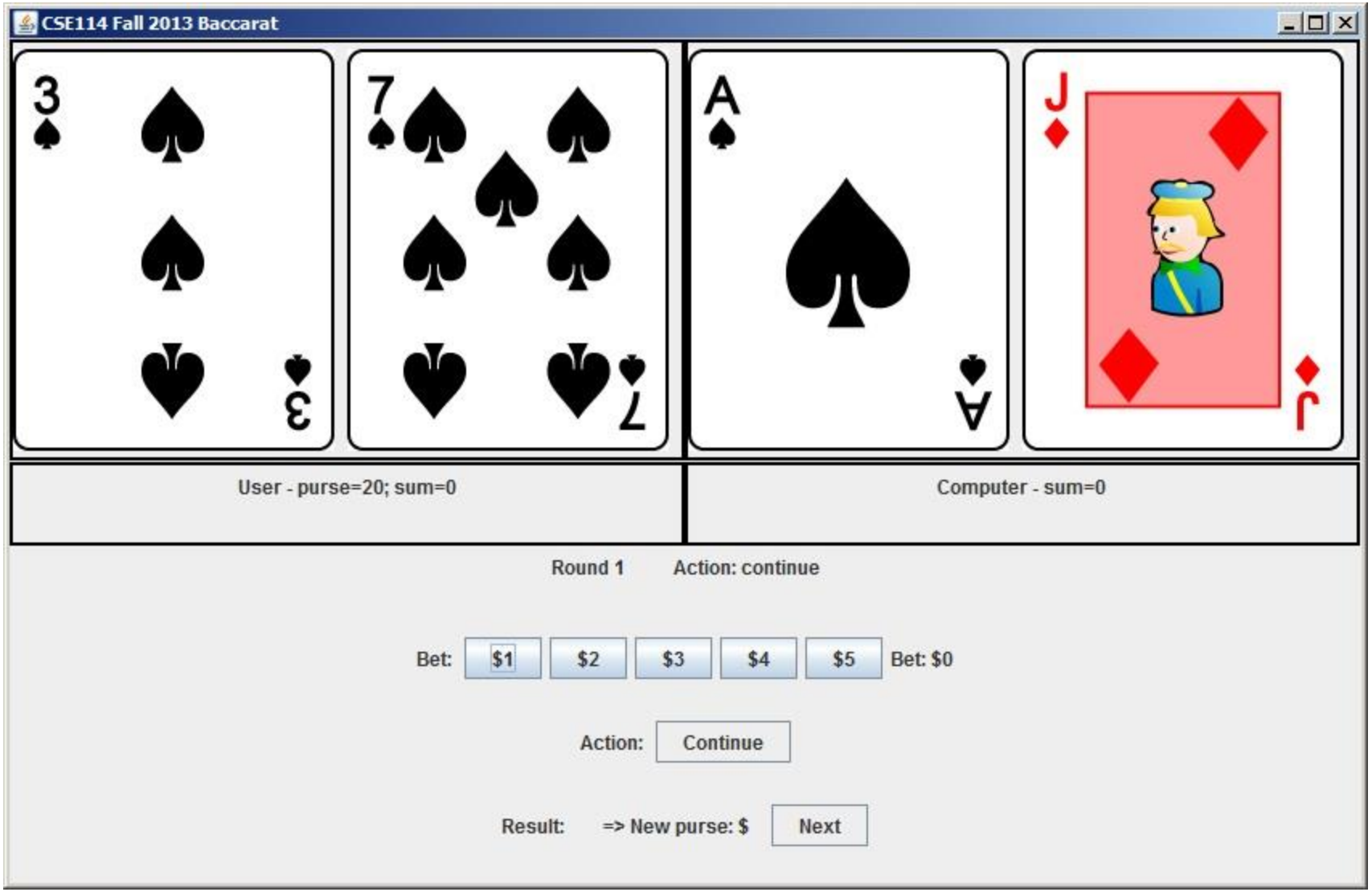


Fall 2012 CSE114 Project: AI Blackjack

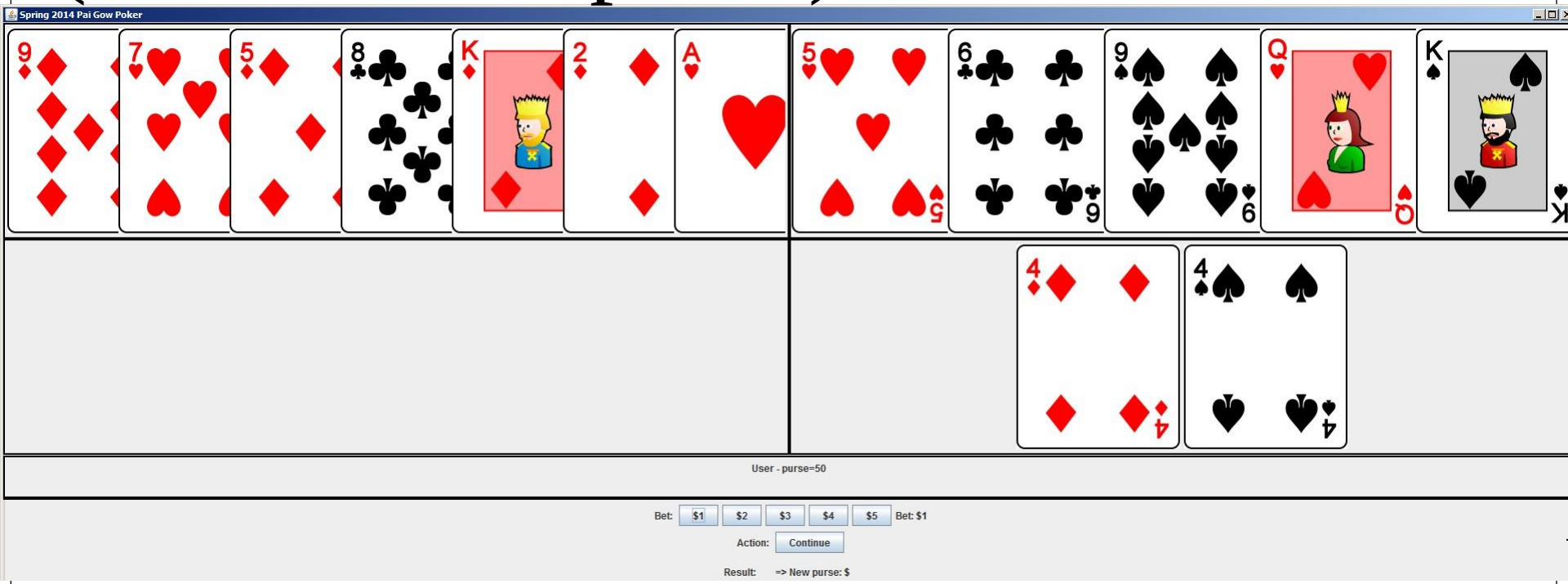



CSE114 Spring 2013 Craps
Chips
48
Pass Line Bet
2
Don't Pass Line Bet
0
Pass Line
Don't Pass Line
$1
$2
$3
$4
$5
Point 10
Good Luck
Roll Dices

Fall 2013 CSE114 Project: Baccarat

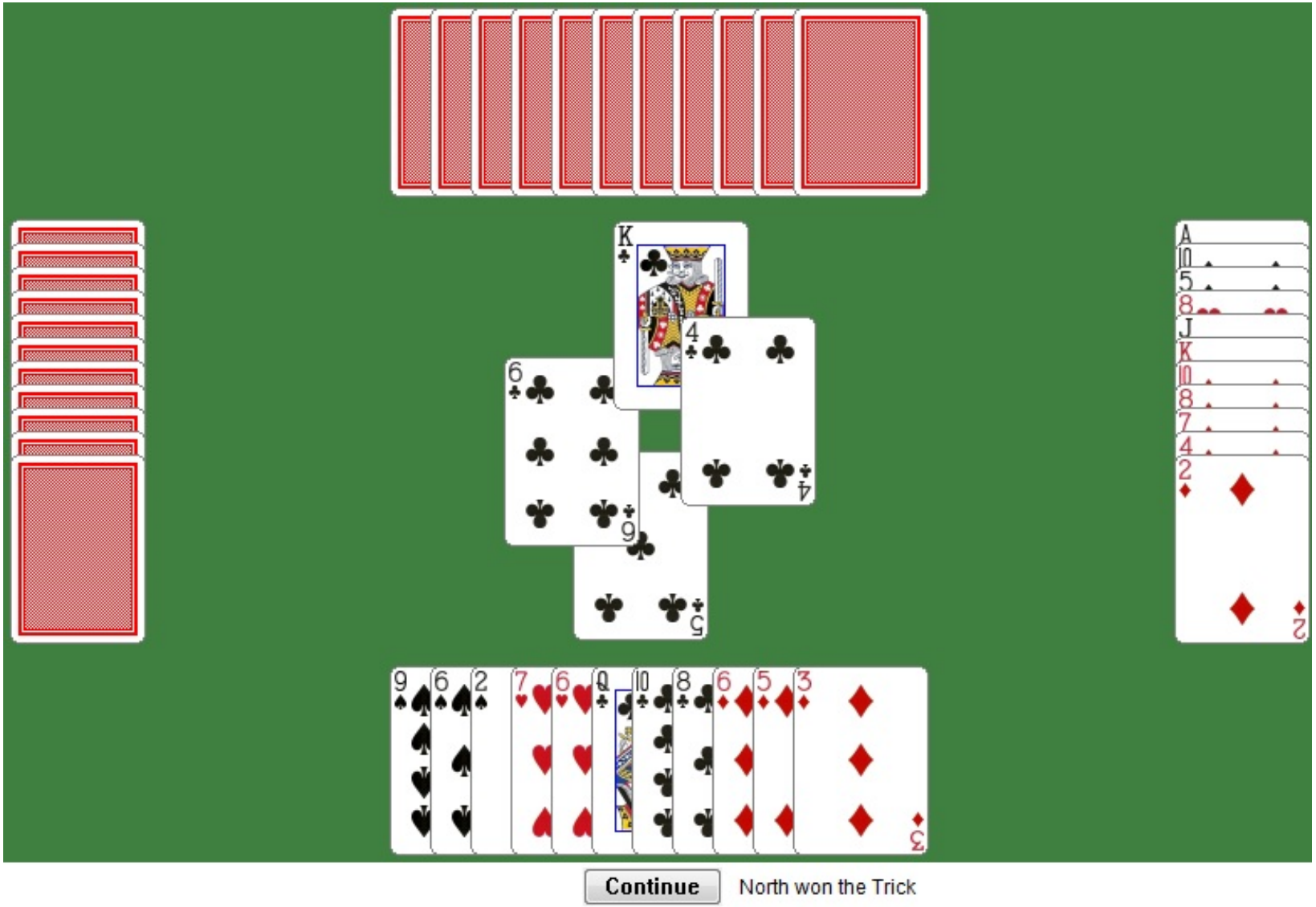


Spring 2014 Project: Pai Gow Poker (double-hand poker)



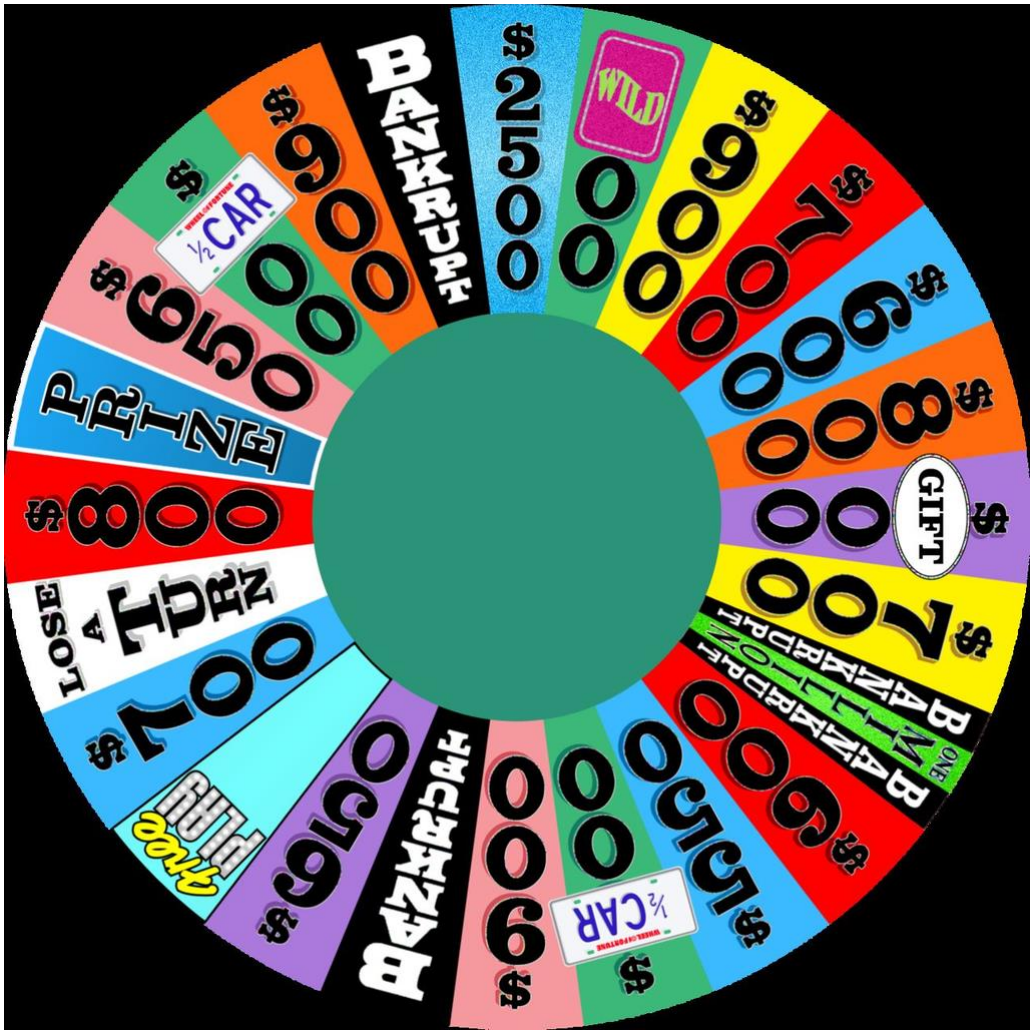
Fall 2014 Project: 24 Game

Spring 2015 Project: Bridge



Summer 2015 Project: Go Fish

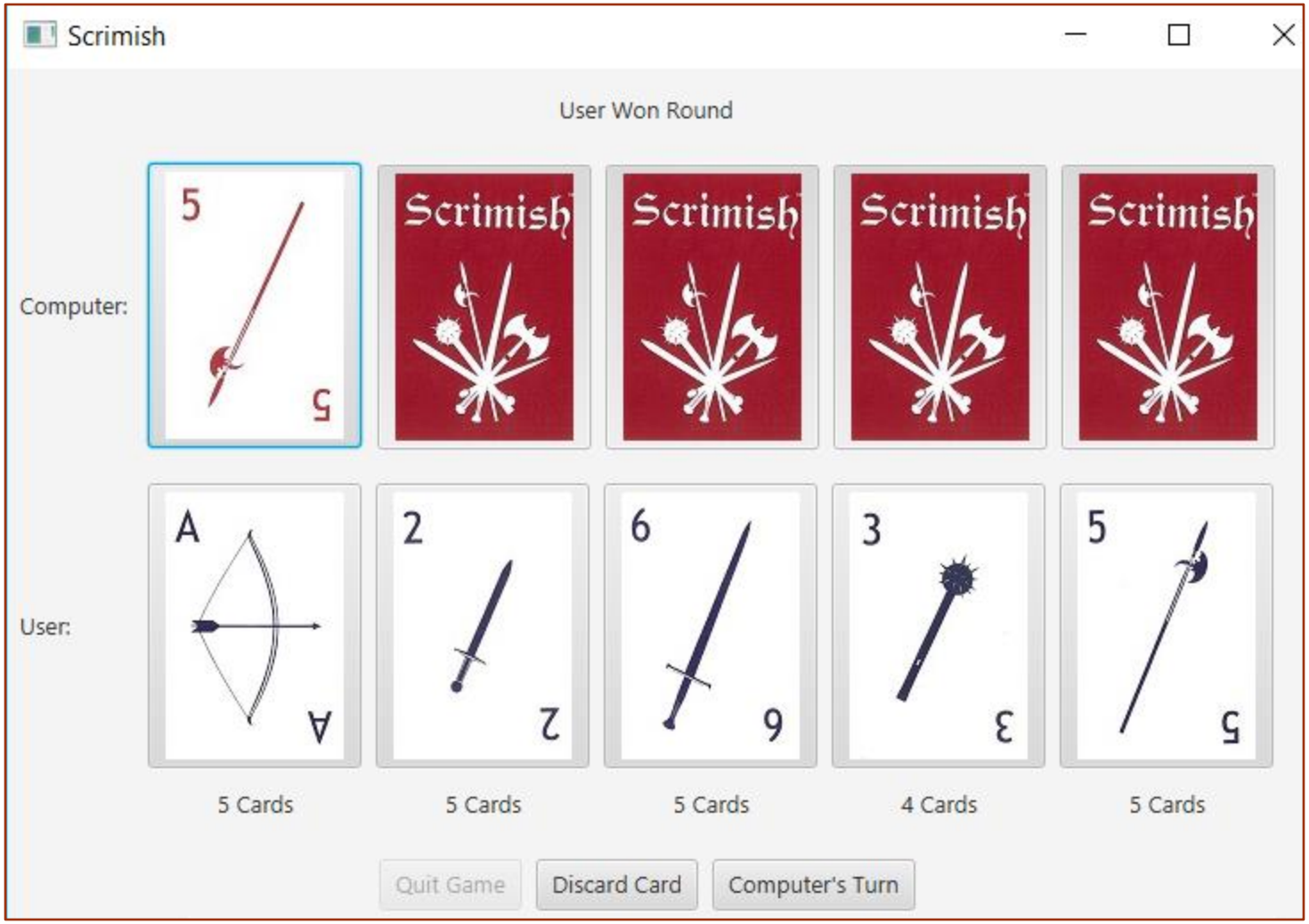
Fall 2015 Project: Wheel of Fortune



Spring 2016 Project: Cosmic Wimpout

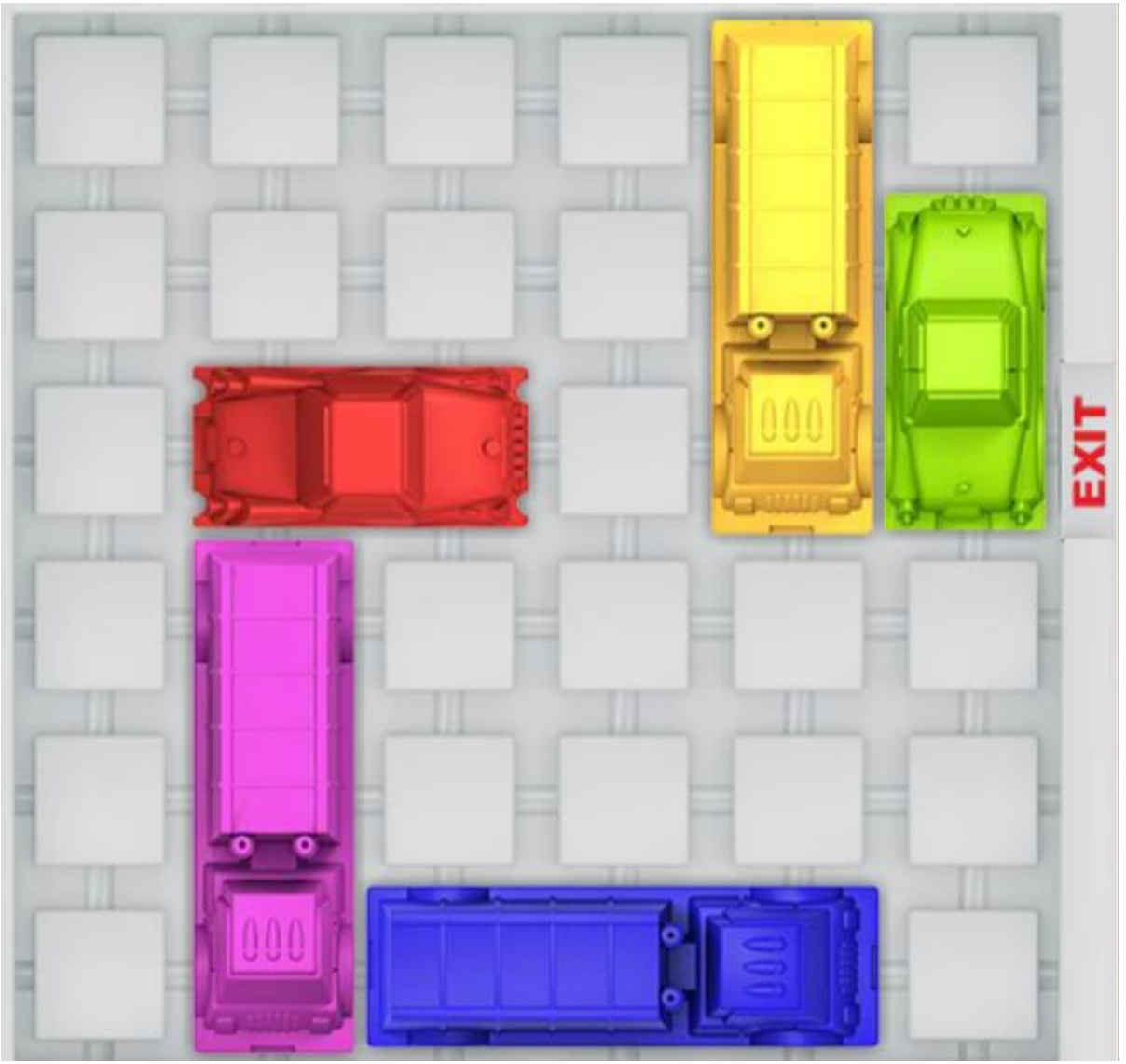
http://cosmicwimpout.com/p/7/How-to-play

Fall 2016 Project: Scrimish

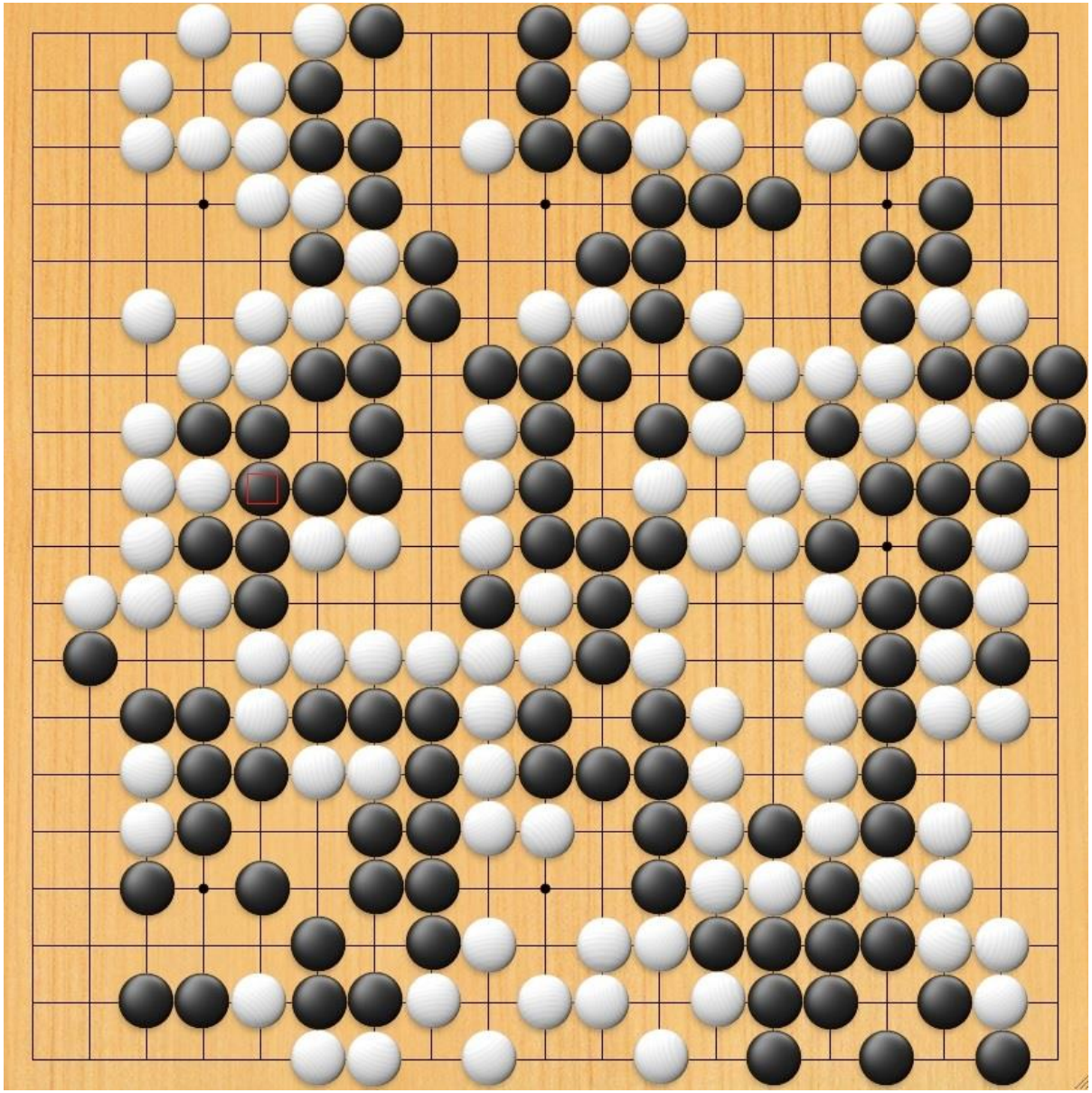


Spring 2017 Project: UNO!

Summer 2017: the rush-hour game



Fall 2016 Project: Go



Tools for Writing Java Programs

- 1st Approach – the bare minimum
  - edit Java source code in text editor (ex: Notepad or Pico)
  - compile source code into class files from command line: javac
  - can be tedious
  - poor interactivity
- 2nd Approach – Integrated Development Environment (IDE)
  - combines writing, compiling, running and debugging Java code into a single application
  - makes coding much more efficient and organized
  - Eclipse, NetBeans, etc.

Java: How does it work?

- Java Source Code
  - you write *??????.java* **files**
- Compile your Program

  `javac ??????.java`

  OR
  - ***Build menu option in the Run menu*** included in the Eclipse IDE
  - The Result is: Java Executable Code (bytecode)
    - *????????.class* files = Java bytecode - not humanly readable
- Now you can run your java program using the Java Virtual Machine (JVM):

  `java YourProgramName`

  OR ***Run*** button included in the Eclipse IDE

Please

- Please be on time
- Please show respect for your classmates
- Please turn off (or use vibrate for) your cellphones

…

- On-topic questions are welcome

Welcome and Enjoy!