

COMPUTER PROGRAMMING IN JAVA

COLUMBIA UNIVERSITY HIGH SCHOOL SCIENCE HONORS PROGRAM

Object Oriented Programming, Part 1

Assignments

2008 Mar 29

Learning to Use Classes: Shapes

A Triangle and a Square class has been provided for you. Both of the classes has a main method, and in the Triangle class it instantiates 3 Triangle objects and prints them out. Do the same for the Square class. Lastly, use the provided Circle class by adding the following data members: a double for the radius, and a String for the color.

Methods

Add a method to each one of our three shape classes that returns the area.

Getters and Setters

Modify the three shape classes so that a circle can only be set to the colors blue, purple, or green, a triangle can only be purple, green, or yellow, and a square can only be green, yellow, or blue.

Constructors

Modify the existing constructor for Triangle so that you can also specify the color. Use this new constructor in the main method to significantly shorten the code. Implement constructors for Square and Circle as well, and apply them to their respective main methods.

Putting it all together: Dicejack

Description: The dice game Dicejack is a variation of casino Blackjack. Consider a heads-up game between a solitary player and the dealer. In each round of the game the following sequence of events occurs:

1. The player rolls a die and accumulates "points" equal to the face value on the die.
2. The player continues to roll until either the player chooses to hold or the player has more than 21 points (busts).
3. If the player busts then the dealer automatically wins the round. If the player holds before reaching 22 points then their turn is done.
4. The dealer now rolls the dice accumulating points in the same way.
5. If the dealer busts (earns more than 21 points) the player wins.
6. If the dealer does not bust then whichever of the two is closer to 21 wins the round, or there is a tie in the case that they have the same score.

Part I: Your first job is to write an application in Java that allows a human user to play this game against a computer dealer. What I haven't told you is how the dealer will decide when to stop. This is called the "dealer's stopping rule". The dealer's stopping rule is basically a point value threshold that once crossed forces the dealer to hold. That is, it's just a number (and it must be constant). For example, suppose the stopping rule is "hold on 17". Then, once the dealer has accumulated 17 or more points they hold, a winner is determined, and the round is over. For the first part of this project you can use any stopping rule you like.

Part Ila: In casino Blackjack the player has a little more information available. After the player gets their first two cards and before they have to choose to hold or get more cards they get to see what one of the dealer's cards are. Include this characteristic of the casino card game into Dicejack.

Part I Ib: In casino Blackjack the player's odds are improved by the fact that if they get a 21 (a "Blackjack"), the player wins 1.5 times the original bet. For Dicejack, we will have a similar concept: if a player hits 21 in exactly 6 rolls, they win 1.5 times the original bet (question: Why exactly 6 rolls? Can you figure out the significance of 6 rolls?)

Part II: Your second job is to determine what the best stopping rule is for the dealer. You will do this by writing a program that plays thousands of games of computer player against computer dealer. Your program will serve as an experiment that allows you to determine the optimal stopping rule via computer simulation.

Part III: Rolling only one dice at a time makes the game a bit easy. Create a variant of Dicejack where on each round the player (or the dealer) rolls two dice at a time.

Bonus: We have arbitrarily set the dealer's stopping rule. Can you figure out a way in which we can systematically test or simulate what the optimal stopping rule for the dealer is? In this context, when we say optimal, we mean "Is there a stopping rule for which we can guarantee a win for the dealer, no matter what stopping rule the player has?".

(This assignment courtesy of CS1004 Spring 2008)