import java.util.ArrayList;

public class UnoGame {

private Deck deck;

private ArrayList<Player> players;

private int currentPlayerIndex;

private Card topCard;

public UnoGame() {

this.deck = new Deck();

this.players = new ArrayList<>();

this.currentPlayerIndex = 0;

}

public void startGame() {

deck.newDeck();

for (int i = 0; i < 4; i++) {

Player player = new Player();

for (int j = 0; j < 7; j++) {

Card card = deck.getCards().remove(0);

player.getHand().add(card);

}

players.add(player);

}

topCard = deck.getCards().remove(0);

}

public boolean isGameOver() {

for (Player player : players) {

if (player.getCardsInHand() == 0) {

return true;

}

}

return false;

}

public void playGame() {

startGame();

while (!isGameOver()) {

Player currentPlayer = players.get(currentPlayerIndex);

Card cardToPlay = currentPlayer.getHand().get(0);

if (canPlay(cardToPlay)) {

currentPlayer.getHand().remove(cardToPlay);

topCard = cardToPlay;

// TODO: implement card effects and turns logic

} else {

Card cardDrawn = deck.getCards().remove(0);

currentPlayer.getHand().add(cardDrawn);

}

currentPlayerIndex = (currentPlayerIndex + 1) % players.size();

}

}

private boolean canPlay(Card card) {

return card.getColor() == topCard.getColor() || card.getValue() == topCard.getValue() || card.getColor() == Card.Color.Wild;

}

}