FEU Institute of Technology
COLLEGE OF ENGINEERING • COLLEGE OF COMPUTER STUDIES

CS ELECTIVE
PARALLEL AND DISTRIBUTED COMPUTING

FINAL PROJECT

1

Centeno, Joaquin Franco Reyes, Jan Andrei Robledo, Ian Justine	Doc. Hadji Tejuco
Name of Student	Name of Professor
06/25/2025	07/02/2025
Date Performed	Date Submitted

Pusoy Clash

Problem Statement

Simulate a digital card draw battle based on the Filipino game **Pusoy Dos**, where **4 players** play through **3 rounds**. In each round, players "draw" a random card (value 2–14). The round winner is the player with the highest card.

Source Code:

```
#include <iostream>

#include <thread>

#include <mutex>

#include <vector>

#include <cstdlib>

#include <chrono>

#include <condition_variable>


using namespace std;


const int NUM_PLAYERS = 4;

const int NUM_ROUNDS = 3;


mutex mtx;

condition_variable cv;

int playsThisRound = 0;

int playersReady = 0;

bool isRoundComplete = false;
```

```
vector<int> drawnCards(NUM_PLAYERS, 0);
vector<int> playerScores(NUM_PLAYERS, 0);

void drawCard(int playerId) {
    srand(static_cast<unsigned int>(time(0)) + playerId);

    for (int round = 0; round < NUM_ROUNDS; ++round)
    {
        this_thread::sleep_for(chrono::milliseconds(300 + playerId * 150));

        int card = (rand() % 13) + 2;
        {
            lock_guard<mutex> lock(mtx);
            drawnCards[playerId] = card;
            cout << "Player " << playerId << " draws a card: " << card << endl;
            playsThisRound++;
        }

        cv.notify_all();
        {
            unique_lock<mutex> lock(mtx);
            cv.wait(lock, [] { return isRoundComplete; });
        }

        {
            lock_guard<mutex> lock(mtx);
```

```
            playersReady++;
        }

        cv.notify_all();
        {
            unique_lock<mutex> lock(mtx);
            cv.wait(lock, [] { return !isRoundComplete; });
        }
    }
}

void determineRoundWinner(int round) {
    int highestCard = -1;
    int winnerId = -1;

    for (int i = 0; i < NUM_PLAYERS; ++i) {
        if (drawnCards[i] > highestCard) {
            highestCard = drawnCards[i];
            winnerId = i;
        }
    }

    playerScores[winnerId]++;
    cout << "\nRound " << round + 1 << " winner: Player " << winnerId
         << " with card " << highestCard << "!\n";
}
```

```
int main() {
    cout << "=== Pusoy Clash Game Simulation ===\n";

    vector<thread> players;
    for (int i = 0; i < NUM_PLAYERS; ++i) {
        players.emplace_back(drawCard, i);
    }

    for (int round = 0; round < NUM_ROUNDS; ++round) {
        {
            unique_lock<mutex> lock(mtx);
            cout << "\n--- Starting Round " << round + 1 << " ---\n";
        }

        {
            unique_lock<mutex> lock(mtx);
            cv.wait(lock, [] { return playsThisRound == NUM_PLAYERS; });
        }

        determineRoundWinner(round);

        {
            lock_guard<mutex> lock(mtx);
            playsThisRound = 0;
            isRoundComplete = true;
        }
```

```
    cv.notify_all();
    {
        unique_lock<mutex> lock(mtx);
        cv.wait(lock, [] { return playersReady == NUM_PLAYERS; });
        playersReady = 0;
        isRoundComplete = false;
    }
    cv.notify_all();
}

for (auto& t : players) t.join();

cout << "\n=== Final Scores ===\n";
int maxScore = -1;
vector<int> winners;

for (int i = 0; i < NUM_PLAYERS; ++i) {
    cout << "Player " << i << ": " << playerScores[i] << " points\n";
    if (playerScores[i] > maxScore) {
        maxScore = playerScores[i];
        winners.clear();
        winners.push_back(i);
    } else if (playerScores[i] == maxScore) {
        winners.push_back(i);
    }
}
```

```
    if (winners.size() == 1) {
        cout << "\nPlayer " << winners[0] << " won with " << maxScore << " points!\n";
    } else {
        cout << "\nDraw between players: ";
        for (size_t i = 0; i < winners.size(); ++i) {
            cout << winners[i];
            if (i < winners.size() - 1) cout << ", ";
        }
        cout << " with " << maxScore << " point(s) each!\n";
    }

    return 0;
}
```

Test Case 1: (Single Winner)

```
=== Pusoy Clash Game Simulation ===

--- Starting Round 1 ---
Player 0 draws a card: 9
Player 1 draws a card: 4
Player 2 draws a card: 10
Player 3 draws a card: 13

Round 1 winner: Player 3 with card 13!
```

```
--- Starting Round 2 ---
Player 0 draws a card: 8
Player 1 draws a card: 6
Player 2 draws a card: 2
Player 3 draws a card: 7

Round 2 winner: Player 0 with card 8!

--- Starting Round 3 ---
Player 0 draws a card: 8
Player 1 draws a card: 2
Player 2 draws a card: 7
Player 3 draws a card: 12

Round 3 winner: Player 3 with card 12!

=== Final Scores ===
Player 0: 1 points
Player 1: 0 points
Player 2: 0 points
Player 3: 2 points

Player 3 won with 2 points!
```

Test Case 2: (Tie / Multiple Winners)

```
=== Pusoy Clash Game Simulation ===

--- Starting Round 1 ---
Player 0 draws a card: 8
Player 1 draws a card: 11
Player 2 draws a card: 5
Player 3 draws a card: 11

Round 1 winner: Player 1 with card 11!

--- Starting Round 2 ---
Player 0 draws a card: 8
Player 1 draws a card: 11
Player 2 draws a card: 12
Player 3 draws a card: 12

Round 2 winner: Player 2 with card 12!

--- Starting Round 3 ---
Player 0 draws a card: 9
Player 1 draws a card: 3
Player 2 draws a card: 4
Player 3 draws a card: 6

Round 3 winner: Player 0 with card 9!
```

```
=== Final Scores ===
Player 0: 1 points
Player 1: 1 points
Player 2: 1 points
Player 3: 0 points

Draw between players: 0, 1, 2 with 1 point(s) each!
```

Test Case 3: (Player wins all rounds)

```
=== Pusoy Clash Game Simulation ===

--- Starting Round 1 ---
Player 0 draws a card: 9
Player 1 draws a card: 9
Player 2 draws a card: 7
Player 3 draws a card: 2

Round 1 winner: Player 0 with card 9!

--- Starting Round 2 ---
Player 0 draws a card: 13
Player 1 draws a card: 7
Player 2 draws a card: 2
Player 3 draws a card: 12

Round 2 winner: Player 0 with card 13!
```

```
--- Starting Round 3 ---
Player 0 draws a card: 13
Player 1 draws a card: 10
Player 2 draws a card: 13
Player 3 draws a card: 5

Round 3 winner: Player 0 with card 13!
=== Final Scores ===
Player 0: 3 points
Player 1: 0 points
Player 2: 0 points
Player 3: 0 points

Player 0 won with 3 points!
```

Tong-its Tagisan

Problem Statement

Simulate a simplified version of **Tong-its** where 3 players take turns **discarding simulated cards** for 3 rounds. The player who discards the highest value each round scores a point.

Source Code:

```
#include <iostream>
#include <thread>
#include <mutex>
#include <vector>
#include <random>
#include <chrono>
#include <condition_variable>

using namespace std;

const int NUM_PLAYERS = 3;
const int NUM_ROUNDS = 5;

mutex mtx;
condition_variable cv;
int playsThisRound = 0;
int playersReadyForNext = 0;
bool roundComplete = false;

vector<int> roundCards(NUM_PLAYERS, 0);
vector<int> scores(NUM_PLAYERS, 0);
```

```
void player(int id) {
    random_device rd;
    mt19937 rng(rd());
    uniform_int_distribution<int> dist(1, 13);

    for (int round = 0; round < NUM_ROUNDS; ++round) {
        this_thread::sleep_for(chrono::milliseconds(300 + id * 150));

        int card = dist(rng);
        {
            lock_guard<mutex> lock(mtx);
            roundCards[id] = card;
            cout << "[Player " << id << "] discards " << card << endl;
            playsThisRound++;
        }

        cv.notify_one();

        {
            unique_lock<mutex> lock(mtx);
            cv.wait(lock, [] { return roundComplete; });
        }

        {
            lock_guard<mutex> lock(mtx);
            playersReadyForNext++;
        }
```

```
        cv.notify_one();

        {
            unique_lock<mutex> lock(mtx);
            cv.wait(lock, [] { return !roundComplete; });
        }
    }
}

int main() {
    cout << "=== Tong-its Tagisan Simulation (5 Rounds) ===\n";

    vector<thread> players;
    for (int i = 0; i < NUM_PLAYERS; ++i)
        players.emplace_back(player, i);

    for (int round = 0; round < NUM_ROUNDS; ++round) {
        {
            unique_lock<mutex> lock(mtx);
            cout << "\n------------------------\n";
            cout << " Starting Round " << round + 1 << "\n";
            cout << "------------------------\n";
        }

        {
            unique_lock<mutex> lock(mtx);
```

```
    cv.wait(lock, [] { return playsThisRound == NUM_PLAYERS; });
}

int maxVal = -1, winner = -1;
for (int i = 0; i < NUM_PLAYERS; ++i) {
    if (roundCards[i] > maxVal) {
        maxVal = roundCards[i];
        winner = i;
    }
}

scores[winner]++;
cout << "\n[Round " << round + 1 << "] Winner: Player " << winner
     << " (Card: " << maxVal << ")\n";

{
    lock_guard<mutex> lock(mtx);
    playsThisRound = 0;
    roundComplete = true;
}
cv.notify_all();

{
    unique_lock<mutex> lock(mtx);
    cv.wait(lock, [] { return playersReadyForNext == NUM_PLAYERS; });
    playersReadyForNext = 0;
```

```
        roundComplete = false;
    }
    cv.notify_all();
}

for (auto& t : players) t.join();

cout << "\n=== Final Scores ===\n";
int maxScore = -1;
for (int i = 0; i < NUM_PLAYERS; ++i) {
    cout << "Player " << i << ": " << scores[i] << " points\n";
    if (scores[i] > maxScore) maxScore = scores[i];
}

// Determine winner(s)
vector<int> winners;
for (int i = 0; i < NUM_PLAYERS; ++i) {
    if (scores[i] == maxScore) {
        winners.push_back(i);
    }
}

if (winners.size() == 1) {
    cout << "\n🏆 Player " << winners[0] << " is the overall winner with "
         << maxScore << " points!\n";
} else {
```

```
        cout << "\n🤝 It's a tie between ";
        for (size_t i = 0; i < winners.size(); ++i) {
            cout << "Player " << winners[i];
            if (i < winners.size() - 1)
                cout << ", ";
        }
        cout << " with " << maxScore << " points each!\n";
    }

    return 0;
}
```

Test Case 1: (One winner)

```
=== Tong-its Tagisan Simulation (5 Rounds) ===

------------------------
 Starting Round 1
------------------------
[Player 0] discards 4
[Player 1] discards 2
[Player 2] discards 4

[Round 1] Winner: Player 0 (Card: 4)

------------------------
 Starting Round 2
------------------------
[Player 0] discards 12
```

```
[Player 1] discards 2
[Player 2] discards 13

[Round 2] Winner: Player 2 (Card: 13)

------------------------
 Starting Round 3
------------------------
[Player 0] discards 12
[Player 1] discards 4
[Player 2] discards 2

[Round 3] Winner: Player 0 (Card: 12)
------------------------
 Starting Round 4
------------------------
[Player 0] discards 12
[Player 1] discards 12
[Player 2] discards 6

[Round 4] Winner: Player 0 (Card: 12)

------------------------
 Starting Round 5
------------------------
[Player 0] discards 5
[Player 1] discards 13
[Player 2] discards 7
[Round 5] Winner: Player 1 (Card: 13)
```

```
=== Final Scores ===
Player 0: 3 points
Player 1: 1 points
Player 2: 1 points

🏆Player 0 is the overall winner with 3 points!
```

Test Case 2: (Tie / Multiple Winners)

```
=== Tong-its Tagisan Simulation (5 Rounds) ===

------------------------
 Starting Round 1
------------------------
[Player 0] discards 4
[Player 1] discards 11
[Player 2] discards 12

[Round 1] Winner: Player 2 (Card: 12)

------------------------
 Starting Round 2
------------------------
[Player 0] discards 13
[Player 1] discards 7
[Player 2] discards 10

[Round 2] Winner: Player 0 (Card: 13)

------------------------
 Starting Round 3
------------------------
```

```
[Player 0] discards 12
[Player 1] discards 12
[Player 2] discards 7

[Round 3] Winner: Player 0 (Card: 12)

------------------------
 Starting Round 4
------------------------
[Player 0] discards 7
[Player 1] discards 13
[Player 2] discards 4

[Round 4] Winner: Player 1 (Card: 13)

------------------------
 Starting Round 5
------------------------
[Player 0] discards 3
[Player 1] discards 8
[Player 2] discards 13

[Round 5] Winner: Player 2 (Card: 13)

=== Final Scores ===
Player 0: 2 points
Player 1: 1 points
Player 2: 2 points

🤝It's a tie between Player 0, Player 2 with 2 points each!
```

Test Case 3: (Player wins all rounds)

```
=== Tong-its Tagisan Simulation (5 Rounds) ===

------------------------
 Starting Round 1
------------------------
[Player 0] discards 6
[Player 1] discards 3
[Player 2] discards 5

[Round 1] Winner: Player 0 (Card: 6)

------------------------
 Starting Round 2
------------------------
[Player 0] discards 12
[Player 1] discards 7
[Player 2] discards 10

[Round 2] Winner: Player 0 (Card: 12)

------------------------
 Starting Round 3
------------------------
[Player 0] discards 5
[Player 1] discards 5
[Player 2] discards 3

[Round 3] Winner: Player 0 (Card: 5)

------------------------
 Starting Round 4
------------------------
```

```
[Player 0] discards 3
[Player 1] discards 3
[Player 2] discards 3

[Round 4] Winner: Player 0 (Card: 3)

------------------------
 Starting Round 5
------------------------
[Player 0] discards 13
[Player 1] discards 11
[Player 2] discards 12

[Round 5] Winner: Player 0 (Card: 13)

=== Final Scores ===
Player 0: 5 points
Player 1: 0 points
Player 2: 0 points

🏆Player 0 is the overall winner with 5 points!
```