

```
<html>
<head>
  <title>歷史覺醒：思想之旅</title>
  <style>
    body { font-family: Arial, sans-serif; text-align: center; background-color: #f0f0f0; }
    #board { display: flex; flex-wrap: wrap; width: 600px; margin: 20px auto; }
    .cell {
      width: 50px; height: 50px; border: 1px solid black; display: inline-block; line-height: 50px; cursor: pointer; background-color:
white;
      position: relative;
      overflow: hidden;
    }
    .player-flag {
      position: absolute;
      top: 5px;
      left: 5px;
      width: 40px;
      height: 40px;
      border-radius: 50%;
      display: flex;
      justify-content: center;
      align-items: center;
      font-size: 20px;
```

```
        font-weight: bold;
    }
    .playerA .player-flag {
        background-color: red;
        color: white;
    }
    .playerB .player-flag {
        background-color: blue;
        color: white;
    }
    .playerA { background-color: white; color: white; }
    .playerB { background-color: white; color: white; }
    .religion { background-color: gray; }
    .history { background-color: gold; }
    .yin { background-color: purple; color: white; }
    #status { margin: 10px; font-size: 18px; }
    #message { margin: 10px; color: darkred; font-weight: bold; max-width: 600px; word-wrap: break-word; }
    #command-input { padding: 10px; margin: 5px; width: 200px; }
</style>
</head>
<body>
    <h1>歷史覺醒：思想之旅</h1>
    <div id="board"></div>
```

```
<div id="status">探索者: 0 | 守舊者: 0 | 回合: 1 | 當前玩家: 探索者 | 圈數: 0</div>
<div id="message"></div>
<div>
  <input type="text" id="command-input" placeholder="輸入命令 (擲骰子/結束/重置/自動)">
</div>
```

```
<script>
  const boardSize = 24;
  const playerA = { pos: 1, wisdom: 0, laps: 0 }; // 探索者
  const playerB = { pos: 1, wisdom: 0, laps: 0 }; // 守舊者
  let currentPlayer = 'A';
  let dice = 0;
  let turn = 1;
  let autoMode = false; // 添加自動模式標誌
  const religionCells = [6, 12, 18, 24];
  const historyCells = [4, 8, 12, 16, 20, 24];
  const yinCells = [3, 9, 15];

  const religionCards = [
    "宗教教條束縛了你的思想，智慧點數-1。宗教的本質是控制，而非解脫。",
    "你被宗教儀式迷惑，智慧點數-2。宗教以儀式掩蓋真相，讓人放棄獨立思考。",
    "宗教假面具蒙蔽你，智慧點數-1。偽善只是控制的工具。",
    "你陷入教派爭端，智慧點數-2。宗教分裂人類，引發無盡衝突。",
```

"宗教規範壓抑你的自由，智慧點數-1。真正的智慧來自質疑。",
"你盲目服從教義，智慧點數-2。盲信阻礙人類進步。",
"宗教戰爭摧毀你的理性，智慧點數-1。歷史證明宗教是爭端的根源。",
"你被宗教恐懼支配，智慧點數-1。恐懼是控制思想的武器。",
"宗教迷信讓你迷失，智慧點數-2。迷信是真相的最大敵人。",
"你被宗教權威壓迫，智慧點數-1。權威掩蓋真相，讓人屈服。"

];

const historyCards = [

"你面臨戰爭與和平的選擇。選擇戰爭：智慧點數-1（短期利益，長期損害）；選擇和平：智慧點數+1（犧牲個人，促進文明）。",

"你發現知識的力量。選擇學習：智慧點數+2（知識是覺醒之光）；選擇忽視：智慧點數-1（無知是歷史的詛咒）。",

"族群衝突爆發。選擇競爭：智慧點數-1（弱肉強食的循環）；選擇合作：智慧點數+1（團結是進步之基）。",

"資源匱乏的困境。選擇掠奪：智慧點數-1（貪婪引發毀滅）；選擇分享：智慧點數+1（共享避免衝突）。",

"你見證歷史重複。選擇反思：智慧點數+1（理解過去才能改變未來）；選擇無視：智慧點數-1（無知重蹈覆轍）。",

"技術進步的抉擇。選擇發展：智慧點數+2（科技推動文明）；選擇保守：智慧點數-1（停滯是衰退之始）。",

"你面對信仰危機。選擇質疑：智慧點數+1（獨立思考是覺醒之門）；選擇盲信：智慧點數-1（信仰蒙蔽真相）。",

"社會制度的誕生。選擇平等：智慧點數+1（公平促進和諧）；選擇壓迫：智慧點數-1（不公埋下禍根）。",

"你發現陰府的痕跡。選擇探索：智慧點數+1（真相超越控制）；選擇逃避：智慧點數-1（逃避讓你受困）。",

"人類自相殘殺的悲劇。選擇停止：智慧點數+1（和平是智慧之果）；選擇參與：智慧點數-2（暴力是歷史的傷痕）。"

];

const yinEvents = [

"陰府調整命運，所有玩家位置後退 3 格。陰府主宰宇宙，凡人無力抗衡其意志。",
"陰府干預歷史，所有玩家失去 1 智慧點數。歷史的循環中，人類不斷重蹈覆轍。",
"陰府施加試煉，所有玩家智慧點數-1。陰府考驗你的意志，覺醒需付出代價。",
"陰府改變規則，所有玩家停止一回合。宇宙主宰無常，人類命運難測。",
"陰府揭示真相，所有玩家智慧點數+1。陰府偶爾顯露真意，但你是否能理解？"

];

```
function initBoard() {  
  const board = document.getElementById('board');  
  board.innerHTML = "";  
  for (let i = 1; i <= boardSize; i++) {  
    const cell = document.createElement('div');  
    cell.className = 'cell';  
    cell.id = `cell-${i}`;  
    cell.textContent = i;  
    if (religionCells.includes(i)) cell.classList.add('religion');  
    if (historyCells.includes(i)) cell.classList.add('history');  
    if (yinCells.includes(i)) cell.classList.add('yin');  
    board.appendChild(cell);  
  }  
  updateBoard();  
  document.getElementById('message').textContent = '請在輸入欄輸入命令：擲骰子, 結束回合, 重置遊戲, 自動';  
  document.getElementById('message').style.color = 'darkblue';  
}
```

```
    document.getElementById('message').style.fontWeight = 'bold';  
}
```

```
function updateBoard() {  
    for (let i = 1; i <= boardSize; i++) {  
        const cell = document.getElementById(`cell-${i}`);  
        cell.className = 'cell';  
        cell.innerHTML = i;  
        if (religionCells.includes(i)) cell.classList.add('religion');  
        if (historyCells.includes(i)) cell.classList.add('history');  
        if (yinCells.includes(i)) cell.classList.add('yin');  
        const existingFlagA = cell.querySelector('.player-flag.playerA');  
        if (existingFlagA) existingFlagA.remove();  
        const existingFlagB = cell.querySelector('.player-flag.playerB');  
        if (existingFlagB) existingFlagB.remove();  
        if (playerA.pos === i) {  
            cell.classList.add('playerA');  
            const flagA = document.createElement('div');  
            flagA.className = 'player-flag playerA';  
            flagA.textContent = '探';  
            cell.appendChild(flagA);  
        }  
        if (playerB.pos === i) {
```

```

        cell.classList.add('playerB');
        const flagB = document.createElement('div');
        flagB.className = 'player-flag playerB';
        flagB.textContent = '守';
        cell.appendChild(flagB);
    }
}

document.getElementById('status').textContent =
    `探索者: ${playerA.wisdom} (圈數: ${playerA.laps}) | 守舊者: ${playerB.wisdom} (圈數: ${playerB.laps}) | 回合: ${turn} |
    當前玩家: ${currentPlayer === 'A' ? '探索者' : '守舊者'} | 骰子: ${dice || '未擲'};
    `;
}

function rollDice() {
    if (dice === 0) {
        dice = Math.floor(Math.random() * 6) + 1;
        document.getElementById('message').style.color = 'darkred';
        document.getElementById('message').style.fontWeight = 'bold';
        document.getElementById('message').textContent = `[${currentPlayer === 'A' ? '探索者' : '守舊者'}] 你擲出了 ${dice}
        點！`;

        movePlayer();
    } else {
        document.getElementById('message').style.color = 'darkred';
        document.getElementById('message').style.fontWeight = 'bold';
    }
}

```

document.getElementById('message').textContent = `[\$ {currentPlayer === 'A' ? '探索者' : '守舊者'}] 本回合已擲過骰子，請先結束回合。`;

```
    }  
  }
```

```
function movePlayer() {  
  const player = currentPlayer === 'A' ? playerA : playerB;  
  let newPos = player.pos + dice;  
  if (newPos > boardSize) {  
    newPos = newPos - boardSize;  
    player.laps++;  
  }  
  player.pos = newPos;  
  handleEvent(newPos);  
  updateBoard();  
  checkGameEnd();  
}
```

```
function handleEvent(pos) {  
  const player = currentPlayer === 'A' ? playerA : playerB;  
  if (religionCells.includes(pos)) {  
    const card = religionCards[Math.floor(Math.random() * religionCards.length)];  
    document.getElementById('message').textContent = `[$ {currentPlayer === 'A' ? '探索者' : '守舊者'}] ` + card;
```



```

    player.wisdom -= card.includes('-2') ? 2 : 1;
  } else if (historyCells.includes(pos)) {
    const card = historyCards[Math.floor(Math.random() * historyCards.length)];
    let choice;
    if (currentPlayer === 'A') {
      choice = confirm(card + "\n 按確定選擇正面選項，按取消選擇負面選項");
    } else {
      choice = false; // AI 選擇負面選項（暴力、競爭等）
    }
    document.getElementById('message').textContent = `[${currentPlayer === 'A' ? '探索者' : '守舊者'}]` + card;
    if (choice) {
      player.wisdom += card.includes('+2') ? 2 : 1;
    } else {
      player.wisdom -= card.includes('-2') ? 2 : 1;
    }
  } else if (yinCells.includes(pos)) {
    const event = yinEvents[Math.floor(Math.random() * yinEvents.length)];
    document.getElementById('message').textContent = `[${currentPlayer === 'A' ? '探索者' : '守舊者'}]` + event;
    if (event.includes('後退 3 格')) {
      playerA.pos = Math.max(1, playerA.pos - 3);
      playerB.pos = Math.max(1, playerB.pos - 3);
    } else if (event.includes('失去 1')) {
      playerA.wisdom--;
    }
  }

```

```

        playerB.wisdom--;
    } else if (event.includes('停止一回合')) {
        turn++;
    } else if (event.includes('+1')) {
        playerA.wisdom++;
        playerB.wisdom++;
    }
} else {
    document.getElementById('message').textContent = `[${currentPlayer === 'A' ? '探索者' : '守舊者'}] 安全通過，歷史的旅程仍在繼續。`;
}
}

```

```

function endTurn() {
    if (dice !== 0) {
        const previousPlayerName = currentPlayer === 'A' ? '探索者' : '守舊者';
        currentPlayer = currentPlayer === 'A' ? 'B' : 'A';
        dice = 0;
        turn++;
        updateBoard();
        document.getElementById('message').textContent = `[${previousPlayerName}] 回合結束，現在輪到 [${currentPlayer === 'A' ? '探索者' : '守舊者'}]。`;
        if (autoMode && currentPlayer === 'B') {

```

```

        setTimeout(aiTurn, 1000); // AI 回合延遲 1 秒執行
    }
} else {
    document.getElementById('message').textContent = `[${currentPlayer === 'A' ? '探索者' : '守舊者'}] 請先擲骰子。`;
}
}

```

```

function checkGameEnd() {
    if (playerA.wisdom >= 10) endGame('探索者因思想覺醒獲勝！');
    else if (playerB.wisdom >= 10) endGame('守舊者因思想覺醒獲勝！');
    else if (playerA.laps >= 3 && playerA.pos === 1) endGame('探索者完成歷史循環獲勝！');
    else if (playerB.laps >= 3 && playerB.pos === 1) endGame('守舊者完成歷史循環獲勝！');
}

```

```

function endGame(result) {
    autoMode = false; // 結束自動模式
    const player = currentPlayer === 'A' ? playerA : playerB;
    let finalMessage = `${result}\n`;
    if (player.wisdom >= 10) {
        finalMessage += "你已覺醒，理解了宗教的假象與歷史的真相。智慧是人類前進的唯一道路。";
    } else if (player.wisdom >= 5) {
        finalMessage += "你雖未完全覺醒，但已開始質疑宗教與歷史的局限。繼續追求真相吧。";
    } else {

```

```
        finalMessage += "你仍深陷宗教與歷史的迷霧。歷史教訓告訴我們：盲信阻礙人類進步。";
    }
    alert(finalMessage);
    resetGame();
}
```

```
function resetGame() {
    playerA.pos = 1; playerA.wisdom = 0; playerA.laps = 0;
    playerB.pos = 1; playerB.wisdom = 0; playerB.laps = 0;
    currentPlayer = 'A';
    dice = 0;
    turn = 1;
    autoMode = false; // 重置時關閉自動模式
    document.getElementById('message').textContent = "";
    initBoard();
}
```

// AI 回合邏輯

```
function aiTurn() {
    if (currentPlayer === 'B' && autoMode) {
        rollDice(); // AI 直接擲骰子
        setTimeout(endTurn, 1000); // 1 秒後結束回合
    }
}
```

```
}
```

```
function handleCommand(command) {
```

```
    const normalizedCommand = command.trim().toLowerCase();
```

```
    if (normalizedCommand === '擲骰子' || normalizedCommand === '擲' || normalizedCommand === 'roll') {
```

```
        rollDice();
```

```
    } else if (normalizedCommand === '結束回合' || normalizedCommand === '結束' || normalizedCommand === 'end') {
```

```
        endTurn();
```

```
    } else if (normalizedCommand === '重置遊戲' || normalizedCommand === '重置' || normalizedCommand === 'reset') {
```

```
        resetGame();
```

```
    } else if (normalizedCommand === '自動' || normalizedCommand === 'auto') {
```

```
        autoMode = true;
```

```
        document.getElementById('message').textContent = '已啟動自動模式，AI 將控制守舊者！';
```

```
        if (currentPlayer === 'B') {
```

```
            setTimeout(aiTurn, 1000); // 如果當前是 AI 回合，立即開始
```

```
        }
```

```
    } else if (normalizedCommand === '幫助' || normalizedCommand === 'help' || normalizedCommand === '?') {
```

```
        document.getElementById('message').textContent = '可用命令：擲骰子, 結束回合, 重置遊戲, 自動';
```

```
    } else {
```

```
        document.getElementById('message').textContent = '無效命令，請輸入 擲骰子, 結束回合, 重置遊戲, 或 自動。輸入
```

```
幫助 或 ? 顯示可用命令。';
```

```
    }
```

```
}
```

```
document.getElementById('command-input').addEventListener('keypress', function(event) {  
  if (event.key === 'Enter') {  
    const command = document.getElementById('command-input').value;  
    handleCommand(command);  
    document.getElementById('command-input').value = "";  
  }  
});  
  
initBoard();  
</script>  
</body>  
</html>
```