**Правила:**

Дошка 8х8 та 24 шашки (12 білих, 12 чорних)

Першими ходять білі

При ході шашка переміщається на незаповнене поле по діагоналі на одну клітинку і тільки вперед

Якщо перед шашкою стоїть шашка протилежного кольору, а поле за нею вільне, то шашка може її побити

Бити можна і назад, якщо є можливість

Якщо після побиття є можливість побити ще одну шашку, то гравець може це зробити

Бити обов'язково (не суперника, а його шашку).

Після того як шашка дістанеться протилежного кінця поля, вона стає дамкою

Дамка може переміщатися на будь-яку кількість клітинок по діагоналі, як уперед, так і назад. Це ж стосується можливості побиття.

Якщо шашка потрапляє на дамкове поле в процесі побиття, то вона негайно ж стає дамкою і рухається далі вже як дамка.

Перемагає в партії той, кому вдалося побити всі шашки і дамки суперника, або коли суперник не може здійснити хід

(усі шашки і дамки суперника, що залишилися, заблоковані шашками іншого гравця).

Якщо впродовж 15 ходів не було здійснено жодного побиття, то гра вважається нічиєю.

Гравці можуть погодитися на нічию раніше, якщо не бачать перспектив перемоги.

**Об'єкти:**

* Білі та чорні шашки
* Білі та чорні дамки
* Поле
* Клітинки для переміщення
* Кнопка звуку
* Повідомлення ходів
* Кнопка повороту поля
* Кнопка пропозиції на нічию
* Кнопка поразки
* Вікно підтвердження
* Повідомлення про результат партії

**Взаємодія:**

* Вікно програми можна згорнути та закрити, але НЕ змінювати в розмірі. Вікно буде мати пропорції 1:1 (з розміром визначитись пізніше)
* При натисненні шашок або дамок позначаються зеленим клітинки поля, на які шашка або дамка може походити (протилежний кінець поля для шашок жовті клітинки)
* При можливості побиття шашки опонента клітинка позначається червоним
* При натисненні цих клітинок шашка або дамка переміщується
* При натисненні кнопки звуку вмикається/вимикається звук гри
* При натисненні кнопки повороту поля вмикається/вимикається анімація повороту поля після ходу гравця
* При натисненні кнопки пропозиції на нічию в наступному ході опонента з'являється вікно з пропозицією
* При натисненні кнопки поразки з'являється вікно підтвердження

**Механіки:**

Для переміщення фігур реалізуємо сітку клітинок

Кожна клітинка міститиме значення, яке відповідатиме за те, якою фігурою вона зайнята

Коли шашка обрана перевіряються 4 клітинки по діагоналі від неї:

* Якщо клітинки немає (шашка знаходиться на краю поля), то вона не позначається
* Якщо на задніх двох клітинках немає шашок опонента або немає змоги їх побити, то вони не позначаються (для кожної індивідуально)
* Якщо на передніх двох клітинках стоїть шашка гравця або шашка опонента, яку не можна побити, то вони не позначаються (для кожної індивідуально)
* Якщо шашка опонента стоїть на будь-якій клітинці (з чотирьох) і її можна побити, то наступна по діагоналі клітинка позначається червоним

Вмикається/вимикається звук гри через натиснення кнопки звуку

При натисненні кнопки повороту поля вмикається/вимикається анімація повороту і вимикається/вмикається різка зміна кута поля

Для вікон підтвердження є кнопки "Так" та "Ні":

* При натисненні "Ні" просто закривається вікно
* При натисненні "Так" закінчується партія

**Assets List:**

Спрайти:

* Біла шашка
* Чорна шашка
* Біла дамка
* Чорна дамка
* Поле 8х8 з дерев'яною рамкою та позначками A-H 1-8
* Зелена клітинка
* Жовта клітинка
* Червона клітинка
* Кнопка
* Закреслена кнопка
* Значок звуку (динамік)
* Значок повороту (поле під кутом зі стрілкою)
* Значок нічиї (дорівнює)
* Значок програшу (білий прапорець або прапорець)

Звуки:

* Хід шашки
* Бій шашкою
* Кінець партії (перемога/поразка)
* Кінець партії (нічия)
* Перетворення в дамку
* Вікно підтвердження

**Rules:**

8x8 board and 24 checkers (12 white, 12 black)

White goes first

When moving, the checker moves to an empty field diagonally by one cell and only forward

If there is a checker of the opposite color in front of the checker, and the field behind it is free, then the checker can beat it

You can also beat backwards, if possible

If after beating it is possible to beat another checker, the player can do it

It is obligatory to hit (not the opponent, but his checker).

After a checker reaches the opposite end of the board, it becomes a king

The king can move any number of squares diagonally, both forward and backward. The same applies to the possibility of beating.

If a checker falls on the king's board during the process of beating, it immediately becomes a king and moves on as a king.

The winner of the game is the one who managed to beat all the checkers and kings of the opponent, or when the opponent cannot make a move

(all the remaining checkers and kings of the opponent are blocked by the checkers of another player).

If no checker has been beaten within 15 moves, the game is considered a draw.

Players can agree to a draw earlier if they see no prospects of winning.

**Objects:**

* White and black checkers
* White and black kings
* Board
* Cell to move
* Sound button
* Notification of moves
* Rotate the board button
* Button for a draw offer
* Defeat button
* Confirmation window
* Notification of the result of the game

**Interaction:**

* The program window can be minimized and closed, but NOT resized. The window will have 1:1 proportions (the size will be determined later)
* When checkers or kings are clicked, the cells of the field on which the checker or king can go are colored green (the opposite end of the field for checkers - yellow cells)
* If it is possible to beat the opponent's checker, the cell is colored red
* When you click on these cells, the checker or king moves
* Pressing the sound button turns the game sound on/off
* Clicking the button to rotate the board turns on/off the animation of rotating the board after the player's move
* When you click the draw offer button, a window with an offer appears on the opponent's next move
* When you click the defeat button, a confirmation window appears

**Mechanics:**

To move the pieces, we implement a grid of cells

Each cell will contain a value that will be responsible for which piece it is occupied by

When a piece is selected, 4 cells diagonally from it are checked:

* + If there is no cell (the checker is on the edge of the board), it is not marked
  + If there are no opponent's checkers on the back two squares or it is not possible to beat them, they are not marked (for each individually)
  + If there is a player's checker or an opponent's checker on the front two squares that cannot be broken, they are not marked (for each individually)
  + If an opponent's checker is on any square (out of four) and can be beaten, the next diagonal square is marked red

Turns the game sound on/off by pressing the sound button

Clicking the field rotation button turns on/off the rotation animation and turns off/on a quick change of the camera angle

There are “Yes” and “No” buttons for confirmation windows:

* + Clicking “No” simply closes the window
  + Clicking “Yes” ends the batch

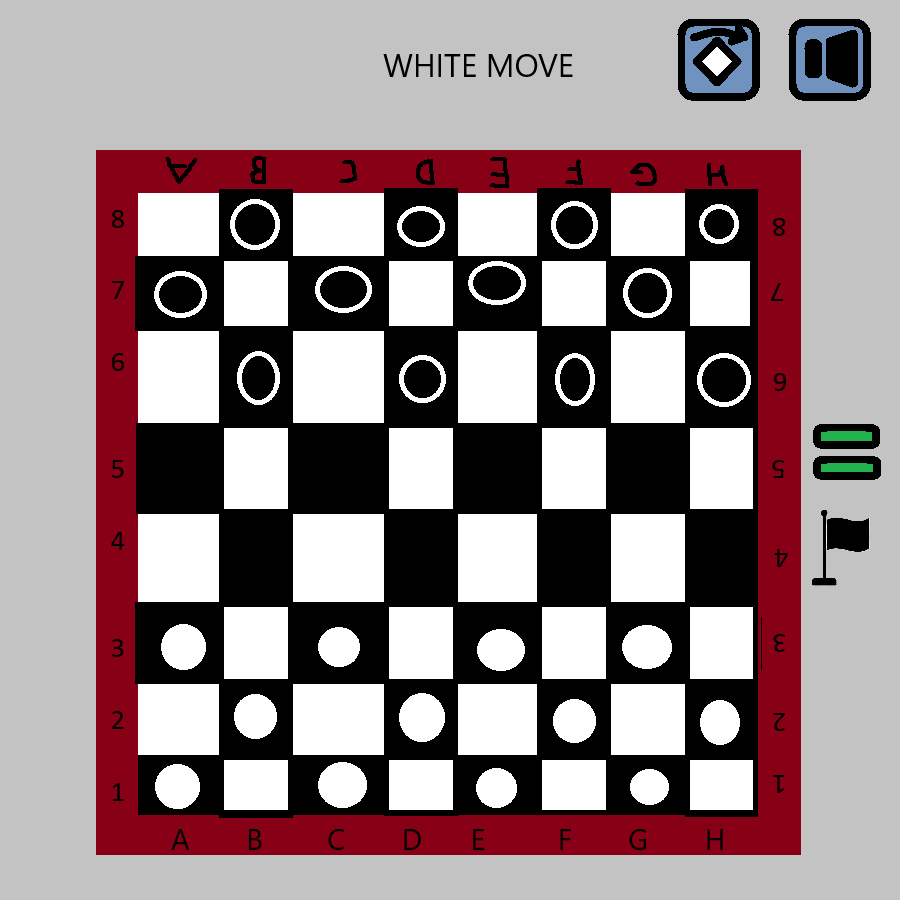
**Assets List:**

Sprites:

* + White checker
  + Black checker
  + White king
  + Black king
  + 8x8 board with wooden frame and markings A-H 1-8
  + Green cell
  + Yellow cell
  + Red cell
  + Button
  + Crossed out button
  + Sound icon (speaker)
  + Turn icon (angled box with an arrow)
  + Draw icon (equal)
  + Surrender icon (white flag or flag)

Sounds:

* + Checker move
  + Capture
  + End of game (win / lose)
  + End of game (draw)
  + Turning into a king
  + Confirmation window



Можливі зміни для Checkers 3D:

- рамка для виділення попереднього ходу (клітинки, на якій стояла шашка)

- кнопка для повноекранного режиму

- більш заокруглений задній фон для вікон (спрайт)

- виділення клітинки, на яку натиснули

[07.12.24]

Створено та оформлено файл GDD.docx, концепт-арт та проект Unity.

[08.12.24]

Створені за допомогою PowerPoint всі необхідні спрайти (для зміни збережено в файлі Sprites).

Спрайт «Board.jpg» та всі звуки знайдені в Інтернеті (джерела вкажуться пізніше).

[09.12.24]

Розміщені всі необхідні елементи гри та створені префаби.

Повідомлення про результат партії змінено на вікно.

Був додатково створений спрайт «Draw Checker.png» для вікна результату партії у випадку нічиї.

Зменшено розмір «Checkers.png» для кращої якості при зменшенні спрайта.

Було вирішено створити клітинки та шашки за допомогою скрипта (чи 1 раз чи кожен раз після запуску гри буде вирішено пізніше).

[10.12.24]

Було створено класи Piece, Cell, BoardManager та CellsCheckersGenerator. Зроблено можливість виділення та відміни виділення клітинок, просування вперед для білих фігур (поки без зміни ходу).

[11.12.24]

Створено клас Checker, який наслідує тепер абстрактний клас Piece.

Клітинка та шашки тепер мають посилання один на одну.

Методи виділення та відміни виділення клітинок розбиті між Piece та BoardManager.

Переміщення фігур тепер виконується в методі всередині класу Piece.

Потрібно виправити спрайт дошки.

Оформлення документації перенесено на завтра (разом з реалізацією побиття фігур).

[12.12.24]

Створено побиття фігур, яке в графічному інтерфейсі виглядає як червоні клітинки.

[14.12.24]

Додана обов’язковість побиття шляхом створення методу, який перевіряє чи є побиття для гравця.

Додана можливість бити декілька разів за хід (якщо це можливо) використовуючи той самий метод.

[15.12.24]

Доданий клас King, який наслідує клас Piece.

Для методів MoveToNewCell та BeatOpponentPiece доданий необов’язковий параметр turnIntoKing, який відповідає за те, чи потрібно на цьому ході перетворювати шашку в дамку.

В класі CellsCheckersGenerator (який перейменовано в CellsPiecesGenerator) доданий метод, який перетворює шашку в дамку.

В класі Cell перевірка ходу, який потрібно здійснити тепер відбувається через змінну string status (можливо потрібно використати інший тип змінної).

[17.12.24]

Дано можливість продовжити побиття після перетворення в дамки за допомогою перевірки на додаткове побиття в методі Start() для King.

Для King додана реалізація GetMoveCells та новий метод, який повертає клітинки відповідного напрямку для ходів.

[18.12.24]

Усі методи/змінні зі словом Beat замінені на Capture

Реалізовано побиття дамкою

Координати клітинок на дошці тепер подані через масив з двох елементів (всі методи, які працювали з цими координатами переписані).

Усі методи які повертали Vector2[] тепер повертають List<int[]> для зручнішого використання масиву координат дошки.

Додано метод CoordsSum() для простішого додавання координат.

[19.12.24]

Виправлена функція Piece.CaptureOpponentPiece().

Додано метод RotateKings() який повертає усі дамки на 180 градусів, щоб корона завжди була правильно повернута.

Доданий метод CheckForMoves(), який перевіряє чи може гравець ходити

Виправлено рисунок дошки (прибрані цифри та букви).

[20.12.24]

Додано звуки для шашок (побиття, хід, перетворення в дамку).

Зроблена анімація повороту дошки (для білих та чорних).

Доданий функціонал для кнопки звуку.

Доданий лічильник для підрахування кількості ходів до нічиї.

Додано метод CheckForEndGame(), який перевіряє чи партії не завершилась.

Метод CheckForMoves() перейменовано в IsMovePossible і він тепер повертає bool.