

Developer A

Sprint3

Developer B

Lav en ny branch: "gameloop_[par]"

I toppen `runGameLoop`, initialiser en variabel, der skal bruges til at tælle runder:

```
int count = 0;
```

Initialiser en boolean variabel, `continueGame` med værdien `true`.

På linjen efter, tilføj et while loop der kører sålænge `continue` er `true`.

Loop'et skal omslutte de sidste linjer i metoden.

Inde i loop'et, har du den linje hvor `currentPlayer` sættes. Ret den så den bruger `count` i stedet for 0.

- På næste linje, tæl `count` op med 1, sådan at `currentPlayer` hele tiden vil pege på den næste spiller i `listen`.

Nu kan koden testes, men hvad sker der når det har været den sidste spillers tur?

- Et sted inde i `while`-loop'et, skal du indsætte et tjek på om `count` er lig med størrelsen på `player` `listen`. Nulstil `count` hvis det er tilfældet. Regn selv ud hvor i kroppen det er bedst at placere dette tjek.

Efter hver spillers tur, skal brugeren have mulighed for at afslutte spillet. Det gør vi ved at prompte brugeren med den nye metode i `TextUI`:

```
continueGame = ui.promptBinary("Fortsæt? (Y/N)")
```

- Indsæt en besked i `endSession` der indikerer at spillet afsluttes.

- Lav metode erklæringer for to nye metoder `throwAndMove` og `landAndAct` i `Game` klassen.

Se *classdiagram_v1* for metodesignaturer.

- Find det sted, hvor vi fortæller hvis tur det er, og flyt linjen fra `runGameLoop` til `throwAndMove`. Kald dernæst `throwAndMove` metoden fra det sted.

- Kør den funktionelle test:

A: Når man starter eller fortsætter et spil, skal der starte et loop hvor turen skifter mellem spillerne indtil man taster "N" til spørgsmålet på om man vil fortætte.

B: Spillet skal afsluttes med en besked, når brugeren vælger IKKE at fortsætte.