User stories:

Title: Game structure

Description: Wall to hold all the tiles. Four players. Players have hands, melds, and ponds.

Estimation: 4

Assumptions: x

Title: Drawing tiles

Description: Players are dealt their initial hands, then take turns drawing tiles

Estimation: 1

Assumptions: x

Title: Discarding tiles

Description: Players take turns discarding tiles from their hands. Everyone can see everyone else's discards.

Estimation: 2

Assumptions: x

Title: Calling tiles

Description: A player can call another player's discard to complete a meld

Estimation: 2

Assumptions: x

Title: Abortive draw - Washout

Description: End the round in a draw if there are no more tiles left to draw

Estimation: <1

Assumptions: x

Title: Text-based display

Description: Display the players' hands and discards through the command line interface

Estimation: 2

Assumptions: x

Title: Text-based input

Description: Players enter their choices by typing in the command line interface

Estimation: 2

Assumptions: x

Title: Abortive draw - 4 kans

Description: End the round in a draw if 4 kans are made, unless they are all made by the same player (in this case, end if a 5th kan is made)

Estimation: <1

Assumptions: x

Title: Abortive draw - 4 riichi

Description: End the round in a draw if all 4 players have declared riichi

Estimation: <1

Assumptions: x

Title: No-ten payment

Description: If the round ends in washout, the players who are not in tenpai must pay points to the players who are

Estimation: 1

Assumptions: x

Title: Tonpuusen game

Description: Hands are played and seats rotate until everyone has been the dealer at least once

Estimation: 2

Assumptions: x

Title: Renchan bonus

Description: If the dealer wins a hand or is in tenpai during washout, he remains dealer and a bonus counter is added to the next hand

Estimation: 2

Assumptions: Assume Tenhou rules for renchan in South 4

Title: Ankan and minkan

Description: A player will be able to make a mikan or ankan from within their hand if they have the necessary tiles

Estimation: 1

Assumptions: x

Title: Tenpai check

Description: A player will be notified if their hand is one tile away from winning (tenpai)

Estimation: 3

Assumptions: x

Title: Riichi

Description: A player will be able to declare riichi if they are in tenpai

Estimation: 2

Assumptions: x

Title: Winning a hand

Description: A player can claim a win if they complete a standard hand (four melds and one pair)

Estimation: 3

Assumptions: x

Title: Wining with non-standard hands

Description: A player can claim a win with the non-standard Seven Pairs and Thirteen Orphans hands

Estimation: 2

Assumptions: x

Title: Furiten

Description: Do not allow a player to claim a win if they previously discarded one of their hand's winning tiles

Estimation: 3

Assumptions: x

Title: Automatic Fu counting

Description: The fu points of a player's hand and melds are calculated upon winning

Estimation: 2

Assumptions: x

Title: Automatic Han counting

Description: A winning hand's han is determined based on the composition of the hand

Estimation: 7

Assumptions: x

Title: Minimum yaku

Description: A player may not claim a win if their hand does not meet the minimum yaku requirement

Estimation: 5

Assumptions: Assume the standard minimum of 1 yaku

Title: Point payments

Description: Points are calculated based on a hand's fu and han, and are paid by the losers to the winner

Estimation: 1

Assumptions: x

Title: Chankan

Description: Allow a player to claim a win by robbing another player's minkan (or ankan for kokushi)

Estimation: 2

Assumptions: x

Title: Nagashi mangan

Description: If a washout occurs and a player's pond includes only terminals and honors, he is awarded points for a nagashi mangan

Estimation: 1

Assumptions: x

Title: Computer player AI basic offense

Description: A CPU player will make choices to form their own hands as fast as possible

Estimation: 14

Assumptions: x

Title: Computer player AI basic defense

Description: A CPU player will evaluate other players' discards and make safe choices to avoid dealing into their hands

Estimation: 10

Assumptions: x

Title: Computer player AI strategy

Description: By evaluating the situation, a CPU player may switch between offense and defense

Estimation: 14

Assumptions: x

Title: GUI: Display hands and ponds

Description: The player's hand and all discard piles will be displayed and updated in the GUI

Estimation: 5

Assumptions: x

Title: GUI: Display round information

Description: The GUI will display the current round wind/number, seat winds, and point totals

Estimation: 1

Assumptions: x

Title: GUI: Choose discard with mouse

Description: The player will select which tile to discard by clicking on it

Estimation: 3

Assumptions: x

Title: GUI: Clickable button prompts

Description: When a player's input is needed for tile calling or other actions, he will be prompted in the GUI and will click on his choice

Estimation: 4

Assumptions: x

Estimation:

Total time needed to finish the game: About 105 days

Game structure = 4

   -Need to create structures to represent all of the data

Drawing tiles = 1

   -Easy, just need methods to remove tiles from wall and add tiles to hand

Abortive draw Washout = <1

   -Easy, just add a check for when the wall has no tiles left

Discarding tiles = 2

   -Need a method for selecting which tile to discard

   -Need a way for players to be aware of each others' discards

Calling tiles = 3

   -Need to check whether or not a tile is callable based on hand tiles

   -After a tile is called, need methods to separate the meld from the hand

Text-based display = 2

   -Need to represent information textually and organize it in a readable way

Text-based input = 2

   -Need to make multiple options menus for different situations

Tenpai check = 3

>-Difficult, needs an algorithm that checks if the hand is one tile away from winning

Winning a hand = 3

>-Most of the work needed here will already have been done with Tenpai check

>-Need to exit the game loop and display endgame information

Wining with non-standard hands = 2

>-Need algorithms that check for these hands

>-Need to add special rules to Tenpai check and Winning to allow these hands

GUI Display hands and ponds = 5

>-Need labels and areas on the GUI to display the tiles

>-Need a way for the GUI to know information about the tiles

>-Need a way for the GUI to know when to update

GUI Display round information = 1

>-Most of the work needed here will be solved in "UI Display hands and ponds"

GUI Choose discard with mouse = 3

>-Need to learn how to use mouse events in Swing

GUI Clickable button prompts = 4

>-Need to add areas on the GUI for the prompts

Abortive draw 4 kans = <1

      -Simple, just need a way to check how many kans a certain player has made.

Abortive draw 4 riichi = <1

      -Simple check, just count the number of riichi

No-ten payment = 1

      -Should be easy to add a check without modifying other areas of the code

Tonpuusen game = 2

      -Need to re-think the flow of the game loop

Renchan bonus = 2

      -Need to research how bonus counters work

      -Bonus will need to be considered in point calculation

Ankan and minkan = 1

      -Need to add extra options during a player's turn

Riichi = 2

      -Need to add extra option during player turn

      -The player turn will need to be handled differently during riichi

Furiten = 3

      -Need to check the discard pile

      -Need to add a check in Winning that disallows claiming the win

Nagashi mangan = 1

      -Easy to add a check without modifying other areas of the game

Chankan = 2

        -Need a way for players to react to another player making a kan

        -Need to add an extra way to claim a win

Automatic Fu counting = 2

        -Need to keep track of some game factors in order to assign values

Automatic Han counting = 7

        -Difficult, need to check hand composition for each of the approx 30 yaku

Minimum yaku = 5

        -Most of the work needed here can be done during Han counting

Point payments = 1

        -Simple calculation, once the fu and han are known

Computer player AI basic offense = 14

        -Needs a lot of planning, needs algorithms for discard choices

Computer player AI basic defense = 10

        -Easier than offense but still complicated

        -Genbutsu, suji, kabe will be easy to start with

Computer player AI strategy = 14

        -Needs to evaluate a lot of factors, and I don't know what those factors will be yet.

Iteration plan:

First Iteration: 15 days

User stories included:

> Game structure = (4)
>
> Drawing tiles = (1)
>
> Abortive draw Washout (<1)
>
> Discarding tiles (2)
>
> Calling tiles (3)
>
> Text-based display (2)
>
> Text-based input (2)

In this iteration, the game consists of taking turns drawing and discarding tiles. Players can call other players' discards to make melds. The events of the game are displayed in a text interface, and player input is taken through the command line. The game ends when the wall runs out of tiles. There is no way to claim a win yet.

Second Iteration: 22 days (This is what I will finish by the end of the semester)

User stories included:

> Tenpai check (3)
>
> Winning a hand (3)
>
> Wining with non-standard hands (2)
>
> Ankan and minkan (1)
>
> GUI Display hands and ponds (5)
>
> GUI Display round information (1)
>
> GUI Choose discard with mouse (3)
>
> GUI Clickable button prompts (4)

In this release, game display and user interaction will take place through a GUI. The user will see images of the tiles on the game table, and will use the mouse to choose a discard or respond to a prompt. Also in this iteration, a player will be able to claim a win if they complete a hand. This includes the non-standard Seven Pairs and Thirteen Orphans hands. Additionally, a player will be notified when they are one tile away from a complete hand (tenpai).

Third Iteration: 20~28 days

User stories included:

        Automatic Fu counting (2)

        Automatic Han counting (7)

        Minimum yaku (5)

        Point payments (1)

        No-ten payment (1)

        Tonpuusen game (2)

        Renchan bonus (2)

May also include:

        Riichi (2)

        Abortive draw 4 kans (<1)

        Abortive draw 4 riichi (<1)

        Nagashi mangan (1)

        Chankan (2)

        Furiten (3)

In this iteration, points for winning hands will be calculated and awarded automatically. The game will also list which yaku the hand has. The game will follow a 4-hand tonpuusen style, and will include several minor win and draw conditions not present in the previous iterations. All main gameplay features should be complete in this iteration.

Fourth, Fifth iterations: ~40 days

User stories included:

        Computer player AI basic offense (14)

        Computer player AI basic defense (10)

        Computer player AI strategy (14)

These iterations will upgrade the opponents into smarter computer players.

Big stories:

"Computer AI" and "Graphical User Interface" were originally big stories. I broke "Computer AI" into three smaller user stories so they would be easier to manage ("offense", "defense," and "strategy"). Similarly, I broke up the original GUI user story into four smaller GUI tasks.

Assumptions:

For "Renchan bonus", I assume that most players are accustomed to the rule that says there is no renchan in South round 4. This is a "house rule," so this may not be the case. The safest way to get rid of this assumption would to add an option that turns the rule on and off.

However, it is a fairly minor rule, so keeping or removing it won't affect the game very much. Because of this, I will likely keep the assumption as a risk (though it is a very minor one).

Git repository: https://github.com/davidsnyder84/MajavaWorking