Abalone problem description

Using a hashmap to represent the board state can aid in fast lookup and processing. It is also easier for humans to keep track of since it only contains possible board positions.

State Representation

- Hashmap with integer keys and string value to represent the board in any state
- Integer key represents the position of the space (ie. A1 is 11, B1 is 21)
- String value will either be “black”, “white”, “empty”
- Movement calculations can be done using key of hashmap
- Have two counters to keep track of knocked out pieces

Initial State

- Standard Layout
  B : 0 - W : 0

```
     w  w  w  w  w
    w  w  w  w  w  w
   e  e  w  w  w  e  e
 e  e  e  e  e  e  e  e
e  e  e  e  e  e  e  e  e
 e  e  e  e  e  e  e  e
  e  e  b  b  b  e  e
   b  b  b  b  b  b
    b  b  b  b  b
```

- Belgian daisy setup

  B : 0 - W : 0

```
      b b w w w
    b b b w w w
  e b b e b b e
e e e e e e e e
e e e e e e e e e
 e e e e e e e e
  e w w e b b e
    w w w b b b
      w w e b b
```

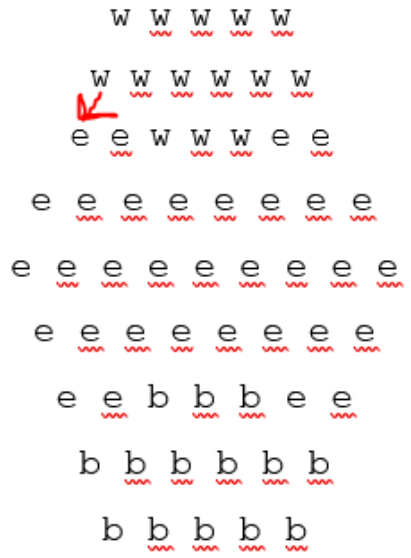
- German daisy

  B : 0 - W : 0

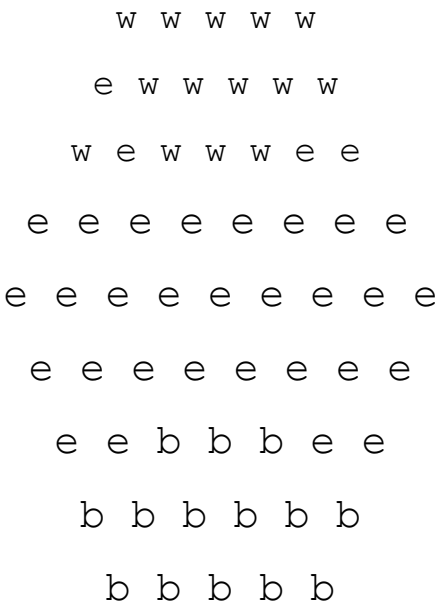
```
      e e e e e
    b b e e w w
  b b b e w w w
e b b e e w w e
e e e e e e e e e
 e w w e e b b e
  w w w e b b b
    w w e e b b
      e e e e e
```

Transition Model

- Each player can move 1, 2, or 3 pieces at a time
- Multiple pieces can be moved in a linear fashion
- Multiple pieces can move side ways (broadside)



White moves from 81 to 71



Goal Test

- Any player having 6 pieces knocked out
- Draw state from a repetition of moves (needs confirming)