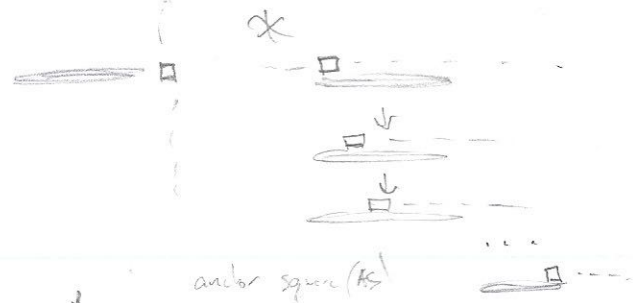
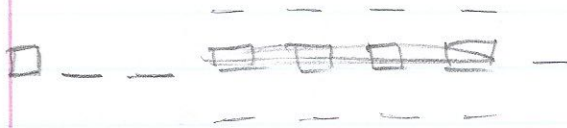
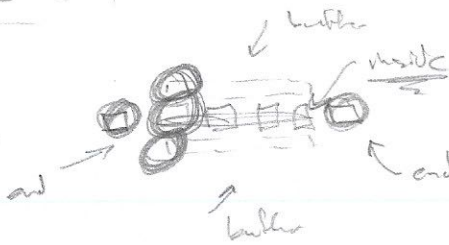


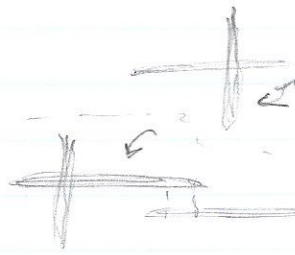
AS - primary AS (\*)

AS - inside - outside  
- head - end  
- buffer



cross checker

Type A (same...) - AS (inside) same direction



(same with vertical...)



opposite direction

Type B - AS (both)

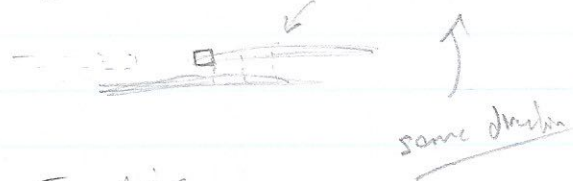


(both)

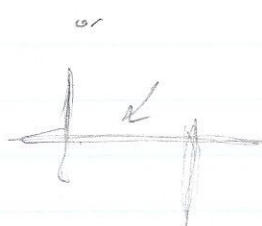
(both)

bats

Type C - AS (outside) "buffer"

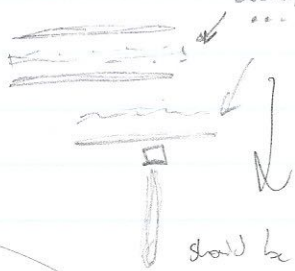


same direction



or

Xenogloss... then there were none (anchor) better overlay...



should be dealt with

cross-checker

Type A & C

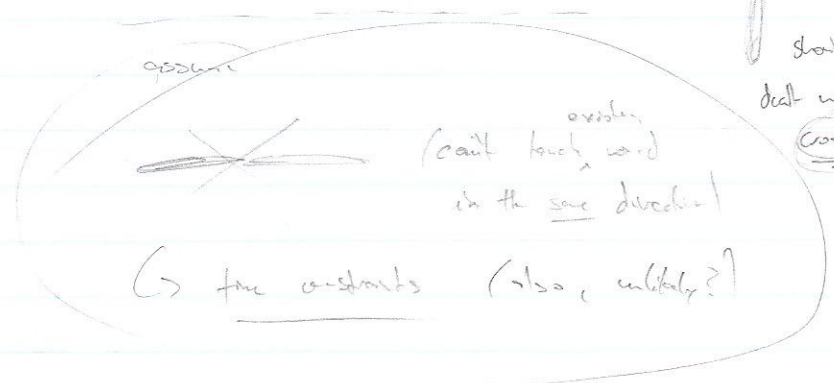


space

right hand has "whistle extension"

while left hand has until the right before the end of a nearby word

only called an "leftmost" or "rightmost" letter to reduce redundant scanning



maybe not...

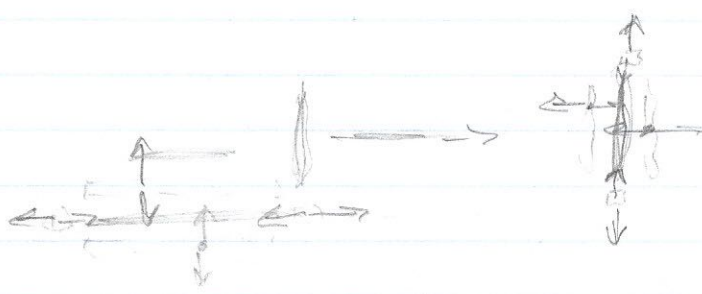
X

# Cross-checks

\* only on outside anchors

\* recursive (find longest "chain" of letters - ~~for~~ opposite direction)

↳ of anchor word generation



\* anchor spaces may be shared → checked by reference word

↳ or checked in both directions

\* Cross checks done after each new word played

↳ baseline value to avoid redundant checking

(only need to

check the "major word")



□ not necessary

recursive

↳ for outside anchors

(until blank space is found)



△ if placed parallel to another word, then

(cross checks added for the immediate board empty squares)

Corners are not needed

(in addition) → "end" first - "left" last - "right"

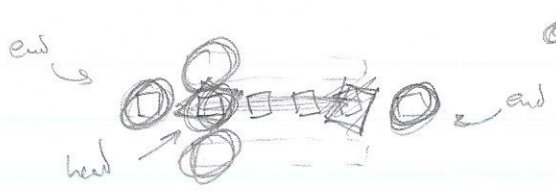
"middle"

⇒ opposite direction

↳ only for anchors

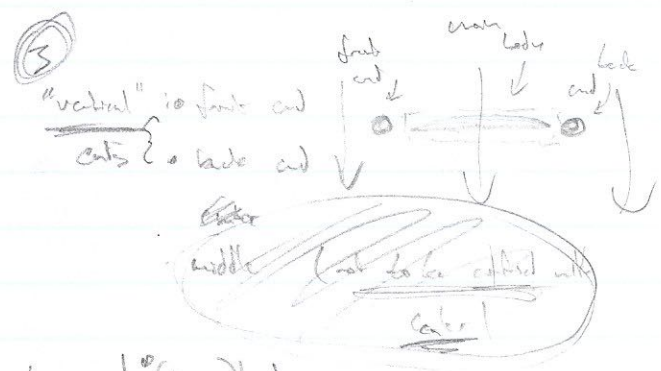
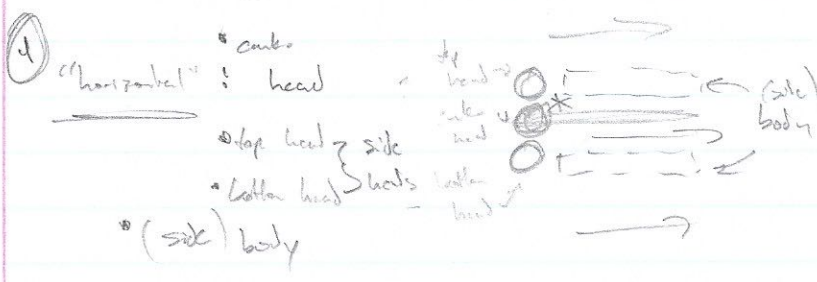


(conducting...)  
 \* Issues with overwriting...



Primary AS (only left - extend on the ...)

Δ left extend (0 - horizontal, all - vertical)

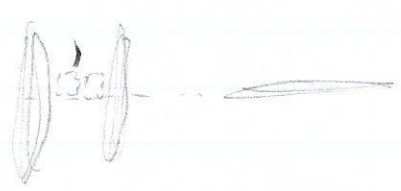


(primary direction...)

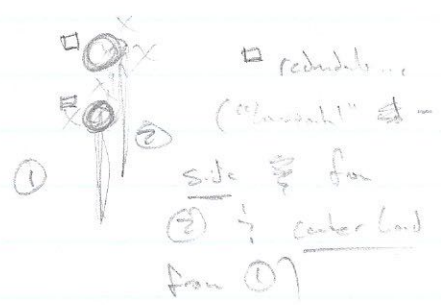
(secondary direction... / (more) body)

\* Construction of words will be left solidly to the center head (primary direction)

Δ only up to "block" square (need space - will not incorporate tiles from other words)



→ covered by "vertical" or center head



\* center head - avoid twice (also in "vertical")

→ just to save, use map { loop: "word" object  
 for possible words value: final paths of word play

→ sort the values (ascending) at the end

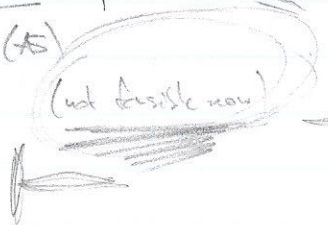
Δ also, not including primary AS

→ still, duplicates are possible... (though, duplicates aren't really the problem... since only the highest score will be used)

including any other anchor source (AS)

→ no need to override equals/hashcode

→ differentiable?



→ if there's a word tile right "before" an anchor (AS)