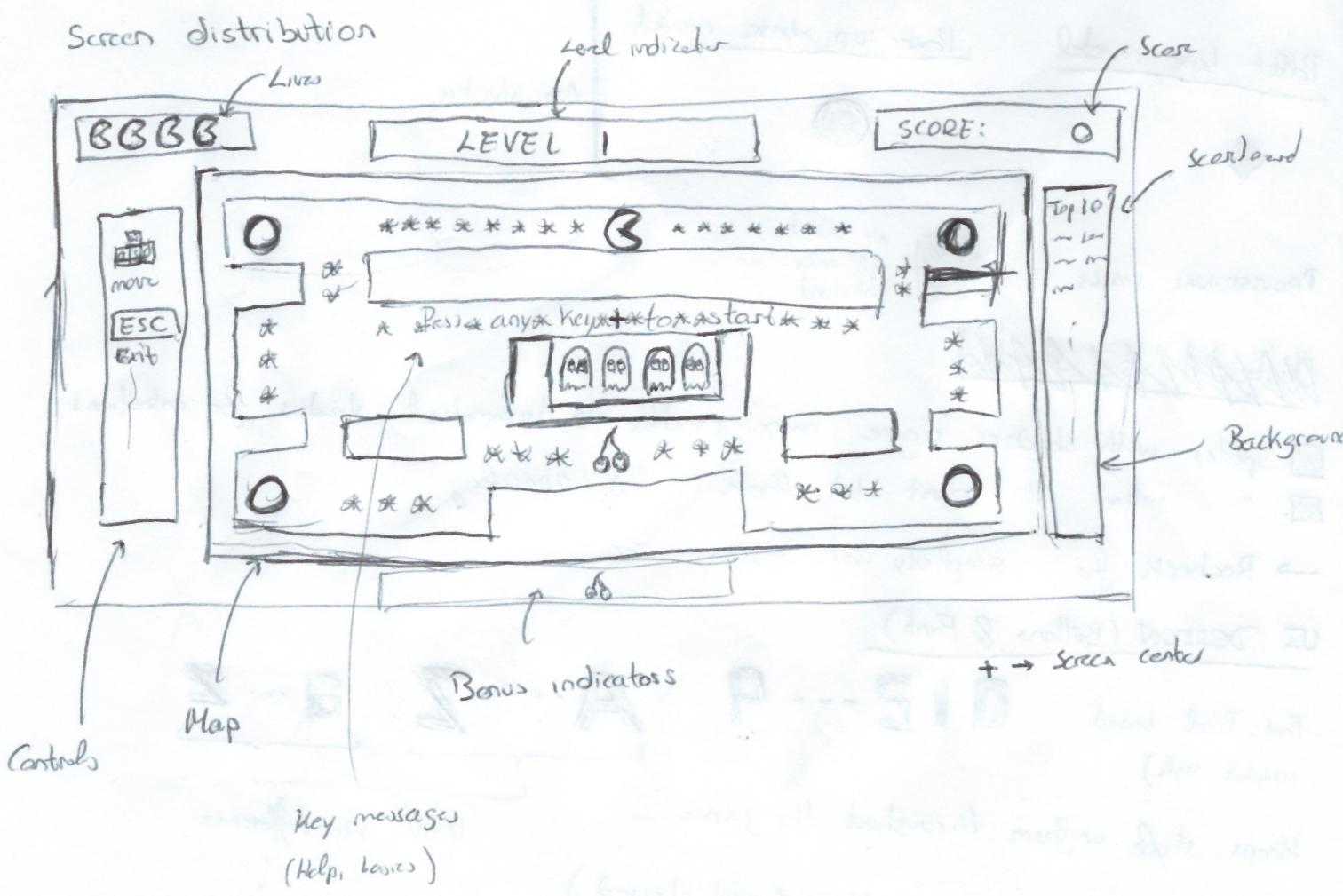


DEV LOG: PacMan

Elements of gameplay

- Pacman
- Ghosts
 - 1 - Red ghost
 - 2 - Blue ghost
 - 3 - Pink ghost
 - 4 - Orange ghost
- Score objects
 - 1 - Pellets
 - 2 - Bonus
 - Pineapple
 - Cherry
 - Watermelon
- Power-ups →

Screen distribution

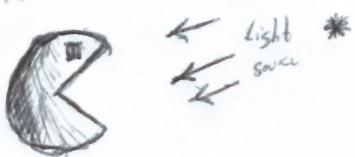


UI elements

- Lives
- Score indicator
- Bonus indicators
 - Cherry
 - Pineapple
 - Watermelon
- Background
- Scoreboard
- Controls ~~→~~
- Messages

SPRITE SKETCHES (16x16 bit)

Pacman base model



Cherry base model



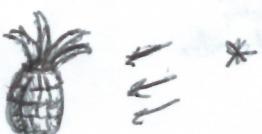
Watermelon base model



Ghost base model



Pineapple base model



Pellet base model



Power-up base model



No shaders

PROVISIONAL WALLS



No border
Fill color
Untextured

~~Provisional walls shaped~~

- Sprites with shaders require more models → Asymmetrical shadows for rotations
 - " without " don't look too appealing
- Recheck for simplicity of design.

UI DESIGN (Buttons & Font)

Font: Pixel based
(Digital style)

0 1 2 ... 9 A ... Z a ... z

Keeps style uniform throughout the game

Visible size difference

Buttons: pointy vs. smooth (pixel-based)



Creates better gradients

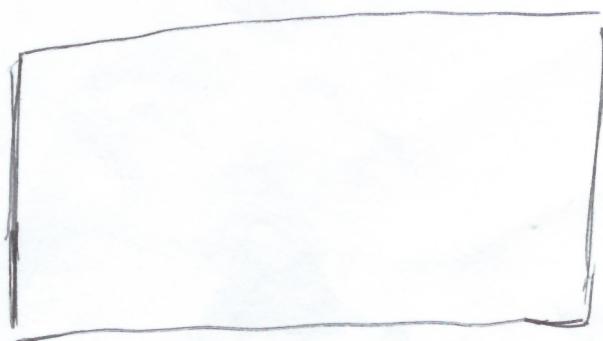
DEV LOG: PackMan

UI Design : Continued

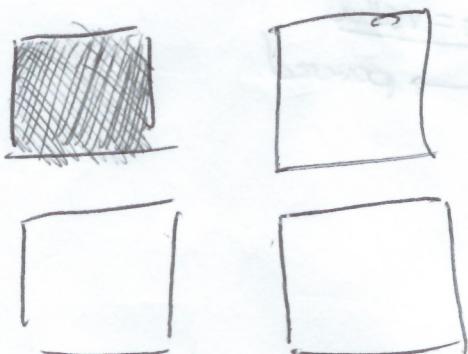
(fridge) blue white button) where state isn't enough for full screen
needs to be longer (left) planning, better screens

→ 0.25 or 0.20
→ Transition

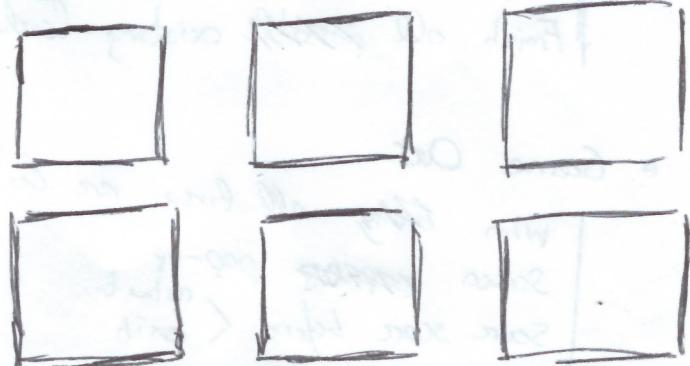
Background & wall design



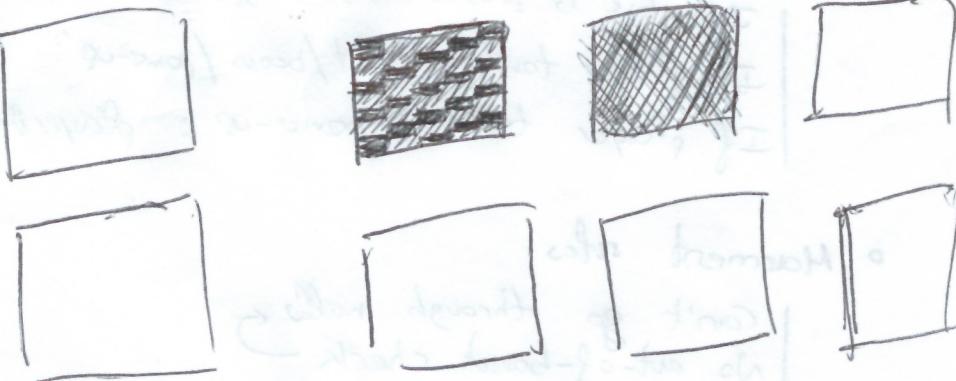
OUTER (UI BG)



INNER (Level BG)

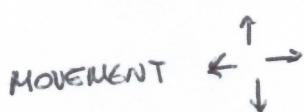


WALLS (generic)



LEVAS WOOD OUT

Player: interface & interactions



WASD vs ←↑→↓

- Player model rotation vs. animated rotation
 - | ROTATION: less art design → Impacts shaded sprites (without shaders and lighting)
 - | ANIMATED ROTATION: Computationally lighter, requires art design

• Controls

ESC: EXIT	W A S D and ←↑→↓ to move
-----------	--------------------------

- Goal
 - | Get all score objects (except for bonus and power-ups)
 - | Finish all ~~existing~~ levels

• Game Over

When ~~taking~~ all lives are lost

Screen ~~displays~~ pop-up

Save score before < exit

• Score & Power-up interaction

If Esc is pressed → save score (not game state) and exit

If player touches pellet/bonus/power-up: score + = value

If player touches power-up: playerMod → powered

• Movement rules

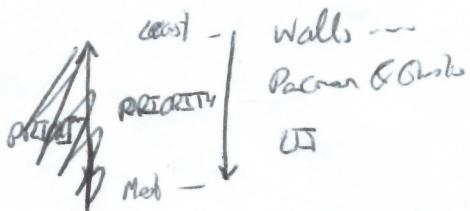
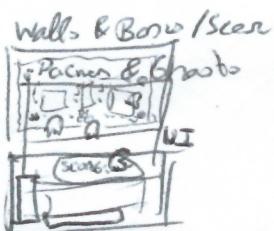
Can't go through walls

No out-of-bounds check

Ghosts can go through each other

Pacman can't go through ghosts (but can go through bonus objects)

SCREEN LAYERS



DEV LOG: Pacman

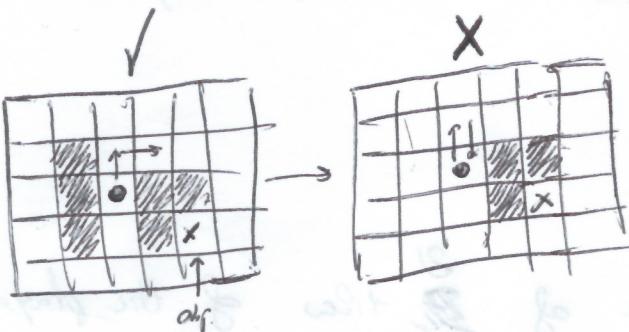
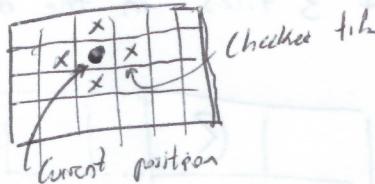
Enemies & AI

4 distinct behaviors

Red
Blue
Pink
Orange

• AI General rules

- AI tries to get closer to its objective on every move
- only checks 1 tile ahead
- It can't turn back

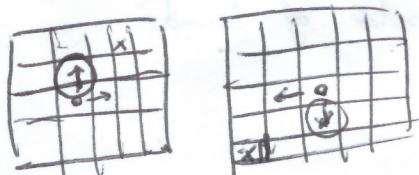
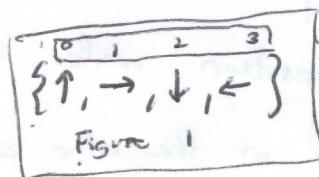


- Avoids loops



■ We will apply clockwise heuristic

- If several movements are possible, the priority will be determined as in Figure 1

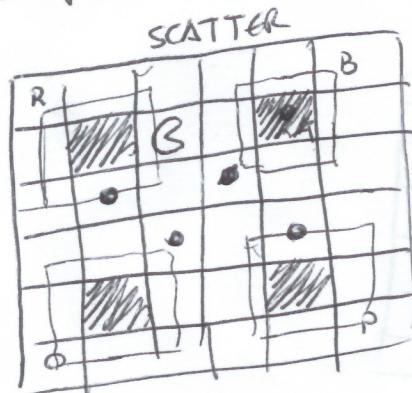
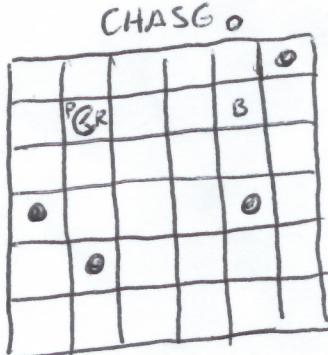


~~Defining ghost behaviors~~
~~Block objects~~
~~Players~~
~~Body parts~~
~~Power~~

CHASE MODE vs. SCATTER MODE

CHASE: The ghosts set their goals to their regular AI ones.

SCATTER: The ghosts go to a fixed point and look nearby (4 corners)

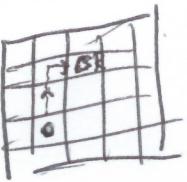


The paths will be the indicated due to the AI rules

Ghost-specific chase behavior

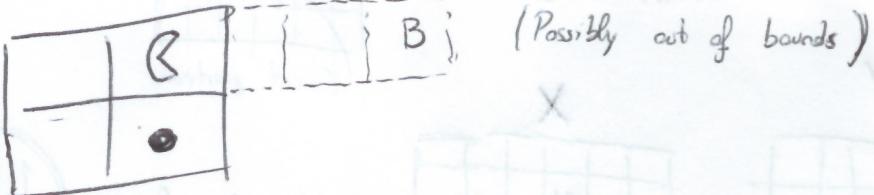
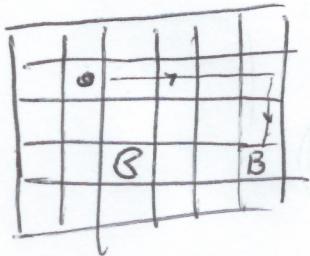
RED GHOST

Goal: Player position



BLUE GHOST

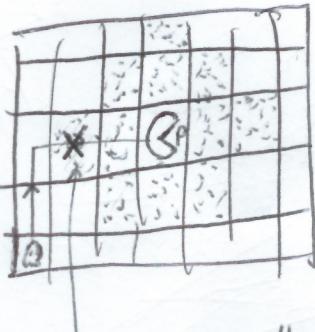
Goal: Player's position + 3 tiles in the direction the player is facing



PINK GHOST

Goal: Player's position while in a range of ~~2~~ tiles of the player
or scatter goal in the other case

subject to
change



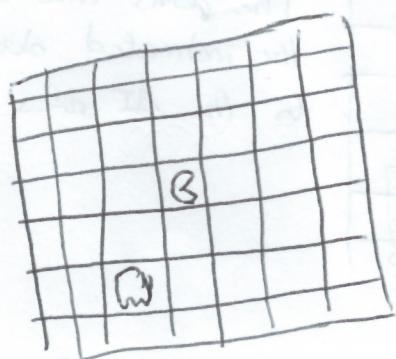
Note: Target possibly out of bounds



Ghost turns to scatter when
it enters the range

ORANGE GHOST

Goal:

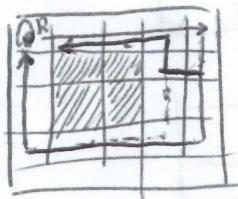


Dev Log: Pacman

AI - Behavior problems

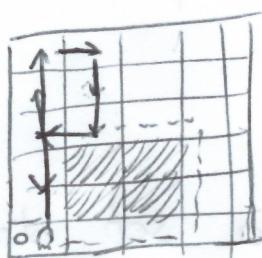
1

Scatter

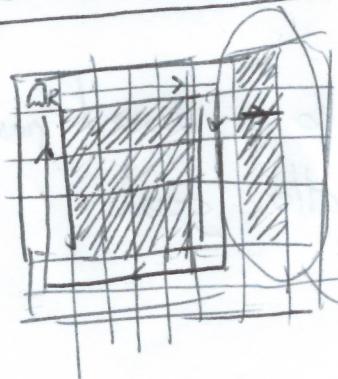


→ Taken route
---> Desired route

Problem with → priority



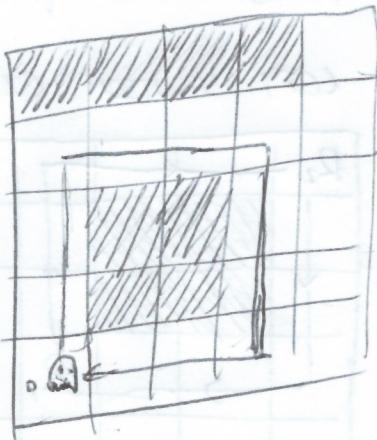
possible solution → Map design choice



Top Left corner

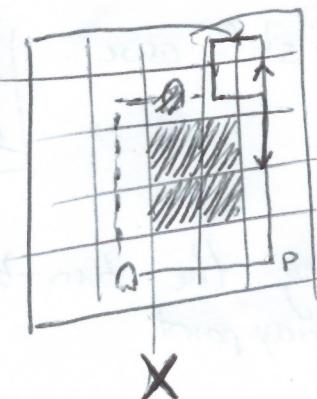
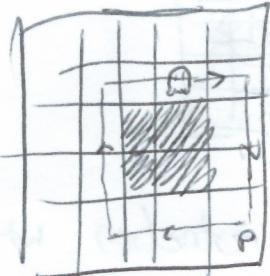
Problem: Unnatural paths

Bottom Left corner



NOTE: This problem can arise if ghosts go through the route in the opposite direction (for

Bottom
right
corner



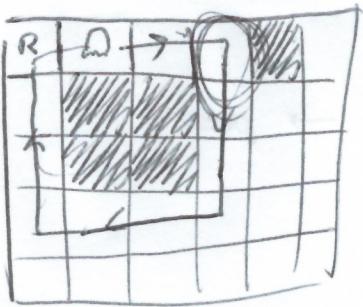
Possible solution? → ADD AI Rule

- If no possible choice decreases dist. to objective → Keep current direction

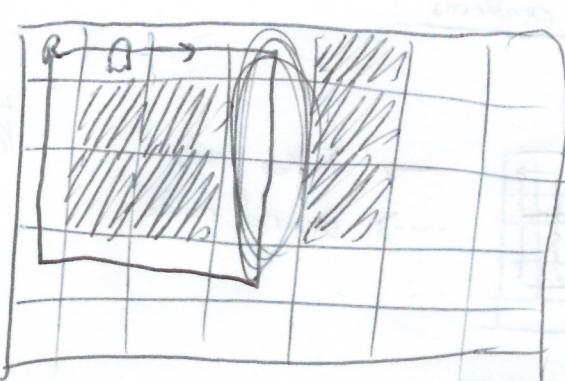
PROBLEM: Paths may still be wrong (TOP RIGHT CORNER CASE)
Maybe combined with map design → Less drastic solutions



Top LEFT solution with modification
in AI also



Top LEFT solution without modification
in AI also

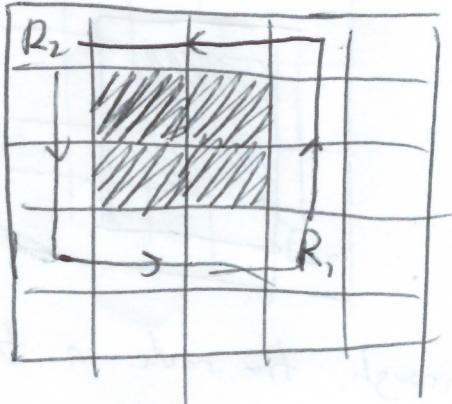


Still creates chokepoints

OTHER SOLUTION: Double to goal

We can force the ghosts to rotate in a set direction in which the problem does not exist

TOP LEFT SOLUTION EXAMPLE

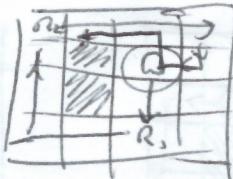


W1: Get the ghost to a desired position

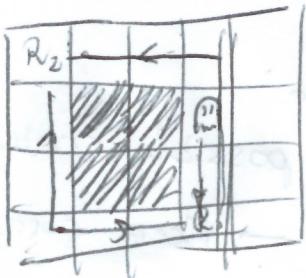
W2: Regular scatter goal

PROBLEM 1: Requires to calculate ghost routes beforehand

PROBLEM 2: The problem may still arise



→ Can be solved eliminating the turn-back rotations when touching the 1st waypoint



at 1st waypoint
rotations



(can move there now)
Takes another
variable. Instead we
right from the bottom