

Game State=1: Direction Screen



Press up, left, right or down to move and avoid the ghost.

If you can make it to the safe zone before running out of lives or time you win!

(click screen to start)



REQUIREMENTS:

- Set Lives=3
- Timer=20 sec
- GameState=1
- This screen must show the directions—though your sprites do NOT have to be visible at this point.
- How will the person switch to Game State 2?

Game State=2: First Level of the Game:



REQUIREMENTS:

- The timer counts down from 20 seconds (or however much time you decide)
- Each time the goblin touches the ghost, it loses a life.
- If the goblin is touching the safe zone **and** there is at least 1 life **and** some time on the clock, then the game state changes to 4.
- If the safe zone has not been pressed before time runs out OR lives run out, then the gameState changes to 3.

Game State=3: If you lose...



REQUIREMENTS:

- Display that you have lost

Game State=4: If you win....



REQUIREMENTS:

- Visual Signal that you have won

Extension: Add Game State=5, which would be level II

** As long as your game has a timer, lives, and gamestates, it can differ from this example however you like, or you may recreate the same game I have shown in the example**

- The user can control the ghost by moving up, down, left or right when the arrow keys are pressed.
- The ghost has two costumes. Switch them accordingly so the ghost doesn't appear upside down.

You MAY want to use some of the following codes:

The image displays four panels of Scratch code blocks:

- Panel 1:** Variable blocks for 'lives':
 - set lives to 0
 - change lives by 1
 - show variable lives
 - hide variable lives
- Panel 2:** An 'if key pressed?' block with 'space' selected, containing a 'then' block with four arrow key options: up arrow, down arrow, right arrow, and left arrow.
- Panel 3:** An 'if touching?' block with 'Ghost1' selected, containing a 'then' block.
- Panel 4:** A collection of control and logic blocks:
 - pick random 1 to 10
 - Comparison operators: <, =, >
 - Logical operators: and, or, not