Algorithms-Introduction

Each object and stage will have certain algorithms applied to it. These algorithms exist in two types, defining algorithms and dynamic algorithms. Every part of the game will have a defining algorithm, as it is form a classification that will control how the object will behave before the player has even interacted with it. Dynamic algorithms apply more to puzzles, these control what happens when the player interacts with the object.

Key:
[]- object
{}-state
mimic {[]}- mimics the properties of an object.
Space- a predetermined distance in pixels.
Animation: - used to show a change in the visuals of the object.
Instance- the section of the game that the user is current interacting with, could be thought as the screen. While the screen might be being showing one level, doesn't mean that there are no other Idle in the background.
None- Removes all associated properties, can be applied to both animations and interactions.
[Player] speak ()- The player will 'say' the text in the brackets (as in it will be entered in a text box above its head).

While every aspect of the game, be it object, character or level transition will have a pseudocode algorithm applied to it, it is also possible to show the algorithms with flowcharts. Below is an example of the algorithm that controls player movement shown as a flowchart:

Player movement: