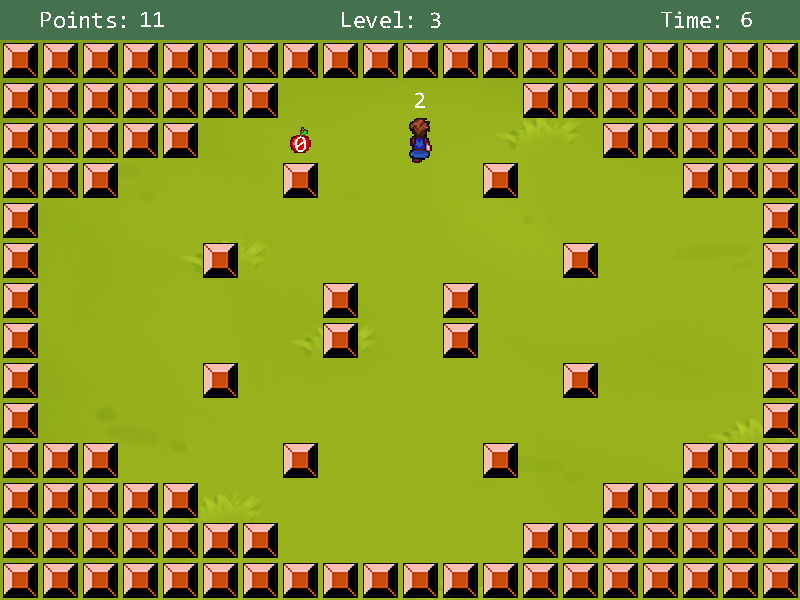
Documentation sheet: you need to complete one of these for your coursework

**Your name/id:** James Simon Morris // 4271058

**Coursework name:** Apple Farmer

**Summary:** Farmer must go around his farm collecting all his apples! In levels 2 and 3 there are caterpillars eating his apples so he has to be quick to collect them all.





**Mandatory (compulsory) requirements: tick them and add a few words of comment if it helps you/us to know what to demo/mark**

1. **Appearance looks OK and appropriate**
   * Appropriate background using image, tile manager used to draw border.
2. **Provide multiple states/stages**
   * 3 levels of game which have different backgrounds using tile manager and file input – moving objects differ on levels 1 and 2/3.
3. **Use the tile manager appropriately – changing at least one tile**
   * Tiles render loaded in image and forms the border/obstacles of the game.
4. **Improve your user-controlled moving object**
   * Moves around using arrow keys/WASD and can also teleport once every 3 seconds using LMB.
5. **Provide multiple different automated moving objects**
   * Apple object that moves to a random location after intersection with player object.
   * Apples object that can only appear once per level worth 3 points, activated by the player pressing RMB.
6. **Provide interaction between moving objects, or a moving object and background**
   * Images drawn making up the border cause the player to lose a life if the player walks into one – interaction between player and tile manager.
7. **Display meaningful changing text on the screen**
   * Text on player (lives) and also on apples showing countdown timer until they disappear and on apple on stages 2/3 showing the same.
8. **Program works well and looks good**
   * Images used for all parts of game // works well.

**Optional requirements – which did you do and what did you do to complete these? Add a few keywords to remind you what to demo.**

1. **Load some data**
   * Below
2. **Advanced data loading**
   * Loads in tiles from txt file that makes up level borders.
3. **Data saving**
4. **Save/load non-trivial state**
5. **Advanced (e.g. animated/scrolling) background**
6. **Animated appearance of user-controlled object and/or automated objects**
   * Player object is animated.
7. **Displayable object images**
   * Apple/apples objects are images.
8. **Creating new displayable objects during the game**
   * RMB to create apples object during game.
9. **Allow user to enter text which appears on the graphical display**
10. **Display text aligned with moving objects**
    * Lives on player object & timer on apple/apples objects.
11. **Complex intelligence on an automated moving object**
12. **Impressive intelligence on an automated moving object**
13. **More complex tile manager interaction**
    * Tiles change depending on the ‘stage’ variable in PsyjsmorEngine class
14. **Implement a hierarchy of moving object classes**
    * PsyjsmorApples is subclass of PsyjsmorApple which is subclass of DisplayableObject.
15. **Non-trivial pixel-perfect collision detection**
    * Apple/apples objects collide with player object.
16. **More complex collision detection**
17. **Really complex collision detection**
18. **Polymorphic state structure**
19. **Implement full pause facility**
    * ESC pauses.
20. **Sellable quality**
    * Please?
21. **Another advanced feature**
22. **A second advanced feature**