My idea

Use this to summarize your idea, plan it using sketches, notes and pseudocode as needed

|  |
| --- |
| - Try to make the game work at this stage.  - Low the expectation.  - Win/ lose condition  - Counting score “Stress level”  - “stress” coming out from the outside of the frame.  - Player tries to hit it to protect the “mind energy” |

Pseudocode.

Declare class ( drawing background, stress, main character and main class)

State float for the speed of the stress.

State int for score to count

Boolean to set the win the condition ( when player misses 5 time);

Arraylist for “the amount of stress” combine with loop function.

Array for “ blurring” in the background.

A couple of yellow papers with drawings on them

Description automatically generated

Where will the inventory skills be demonstrated? List every single one to be sure you’ve included them.

|  |
| --- |
| Shapes (1. line, ellipse, rect, triangle, quad, arc, curve 2. fill, stroke, strokeWeight, noFill, noStroke, color 3. Modes: CORNER, CORNERS, CENTER, RADIUS)  🡺 Draw the background.  System 4. setup(), draw() use the run in the main class  5. random(), create random position where stress appear 6. constrain() keeping the score from 0 to 10 only  7. Use KeyPressed to move the main character.( AWSD) 8. increment operators: ++ for the number of “stress” coming out and score count. 9. declare and use a local variable (?) set the variable and use it in the function (like declare float d then use d= dist(a,b,x,y) or anything else (random) to set the function void draw()) 10. declare and use a global variable (?) declared outside any function, and they can be accessed (used) on any function in the program.( create new class, then mention the new class into main class by  Debugging 11. println() write the note to show player how to play. Control flow 12. conditional statements: if, else if, else: will apply to make win/ lose condition. 13. Boolean expressions: == to make true of false statement in “if”, >= for win condition., 14. Logical operators: && combine 2 condition to make true of false statement. 15. switch statement (?) Loops 16. for loop, make the stress coming back once they get out of the frame 17. A nested loop (?) 18. break() can be used but don’t know yet  19.Loop will repeat the object with the known number of iteration, while loops will continue repeat until the program is proved wrong. Functions 20. Declare & call a function with no parameters and no return type 21. Declare & call a function with a return type 22. What’s the difference between parameters and arguments? 23. Pass by copy (value): declare and use a function that takes int, float, char, etc as an argument 24. Pass by reference (objects): declare and use a function that takes an object as an argument Classes/objects 25. Class contains object and can use multiple times with the shorter line of code.  26. What is a constructor function? What does it do and when? 27. Why should each class have its own tab in Processing? 28. Write a class with a constructor function 29. Use the keyword new to instantiate an object 30. Write a constructor function with parameters.  Lists 31. Array is fixed size while the Array list can grow and resize as needed. 32. Why would you want to go through a list backwards, decrementing the index? 33. Initialize and populate an array 34. Initialize and populate an ArrayList 35. Manage a set of objects with an array or ArrayList 36. Use an ArrayList method: size(), get(), remove(), contains() Vectors 37. When should you use PVector instead of float variables? 38. PVector for Stress position and velocity 39. Do some basic physics: use position, velocity, and acceleration (due to gravity) vectors 40. Find the direction and distance between two points 41. Create a random 2D vector 42. What is a normalized vector, why is it useful? 43. Using the Processing documentation look up a method in the PVector class that’s new to you and use it in your code. |
| REMEMBER TO NOTE EVERY INVENTORY SKILLS NEXT TO THE CODES FOR THE MARKSSSS |

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone 1** | **Milestone 2** | **Milestone 3** | **Milestone 4** |
| What will I deliver?  Finish the background and needed function like score count display .  Win display and lost display | Adding the function | You are strongly encouraged to deliver your finished game at Milestone 3. |  |
| Which inventory skills will this demonstrate? List them.  1.Lines, ellipse, rect, triangle | 1. i++ for stress amount  2. Adding Array for Stress  3. Score count function  4. Adding Array list for buzzy feelling, which makes the distraction. | 1. Lose condition  2. If have time add, winning condition too.  3. Adding instruction by printl();  4. using break () here ? | 1. Touching up . 2. Hopefull to finish up by milestone 3. |
| 2. boolean | 5. PVector for stress position and velocity | 5. Remove () for every buzzy distracting when get out of the screen |  |
| 3. class. | 6. String action for animation player | 6. add Losing screen background. |  |
| 4. Array | 7. random() that the stress will come out random at the bottom from the straight line. | 7. using switch for different emotions when keypressed. |  |
| 5. Keypressed | 8. for loop, make the stress coming back once they get out of the frame | 8. Constrain() the wideness of stress can go in from the bottom (50 to 350 only) |  |
| 6.PVector | 9. |  |  |
| 7. setup () | 10. list of float ,int for meausuring | 9. Set Enter to refresh the game |  |
| 8.draw() |  | 10.== for setting the keypressed A and W to true and false for release |  |
| 9. && |  |  |  |
| 10.if () else{} |  |  |  |
|  |  |  |  |
|  |  |  |  |
| You should deliver approx. 10 skills at this milestone | You should deliver approx. 10 skills at this milestone | **You must deliver 30 inventory skills by this milestone.** |  |