Evan Daveikis: My idea

Info:

GitHub Username: Tobogganeer

Repo Link: https://github.com/Tobogganeer/PEntityExtraction

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

From the start, I knew I wanted to make a game for the arcade cabinets. After looking at various options (seen in the paper notes below), I decided on a digital version of a card game I created for our Foundations of Game Design class. It features a variable number of players working cooperatively to clear a facility of anomalous entities. It was inspired thematically by cooperative SCP games I have played before, and the gameplay started with my love of procedurally generated content. The game is turn based, with players picking up cards and taking actions on each turn.
My idea was that you could have numerous players playing on one arcade cabinet, using one stick to take actions and another to pan around the map. I haven't seen any board games done on arcade cabinets before, so I think it will be a unique idea! (mind you, I haven't seen many arcade games in general)
See further planning in the images in the zip and the rules document included (for the board game rules)

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

Shapes	------------------------------
1. line, ellipse, rect, triangle, quad, arc, curve	Drawing cards to the screen
2. fill, stroke, strokeWeight, noFill, noStroke, color	Used to colour cards
3. Modes: CORNER, CORNERS, CENTER, RADIUS	Used when drawing certain shapes
System	------------------------------
4. setup(), draw()	Always included
5. background(), random(), noise()	Used to draw cards and clear the screen
6. constrain(), dist()	Used to find closest player

7. keyPressed(), keyReleased(), keyPressed, mousePressed(), mousePressed	Used to detect keypresses for keyboard play (game will be on arcade cabinets)
8. increment operators: ++, +=, --, -=, *=, /=	Used in many places (hp, etc)
9. declare and use a local variable	Used a bajillion times
10. declare and use a global variable	Used for initializing controller library, storing board and players, etc
Debugging	----------
11. println(), stop()	Used for debug stats and error messages
Control flow	
12. conditional statements: if, else if, else	Used an infinite number of times
13. Boolean expressions: ==, >=, <=, >, <, !=	^^^
14. Logical operators: &&, ||	^^^
15. switch statement	Will be useful for use with enums (directions, moving, etc)
Loops	----------
16. for loop, while loop	Millions of uses
17. A nested loop	Checking if any player has a card (loop through all players, and all cards of each player) and locking doors
18. break()	Pretty sure this shouldn't be a function lol, and also same use case as the previous one + level generation
19. What's the difference between a for loop and a while loop?	-- Answered in questions doc --
Functions	----------
20. Declare & call a function with no parameters and no return type	handleInput(), drawLevel(), infinite possiblities
21. Declare & call a function with a return type	Various util and math functions
22. What's the difference between parameters and arguments?	-- Answered in questions doc --
23. Pass by copy (value): declare and use a function that takes int, float, char, etc as an argument	Math and util functions

24. Pass by reference (objects): declare and use a function that takes an object as an argument	Any function that takes a player, card, or PVector
Classes/objects	----------
25. What's the difference between a class and an object?	-- Answered in questions doc --
26. What is a constructor function? What does it do and when?	-- Answered in questions doc --
27. Why should each class have its own tab in Processing?	-- Answered in questions doc --
28. Write a class with a constructor function	Every class
29. Use the keyword new to instantiate an object	Billion uses
30. Write a constructor function with parameters	Almost every class, I imagine
Lists	----------
31. What's the difference between an array and an ArrayList?	-- Answered in questions doc --
32. Why would you want to go through a list backwards, decrementing the index?	-- Answered in questions doc --
33. Initialize and populate an array	Board tiles
34. Initialize and populate an ArrayList	Players, cards
35. Manage a set of objects with an array or ArrayList	^^^
36. Use an ArrayList method: size(), get(), remove(), contains()	^^^ + in a loop
Vectors	----------
37. When should you use PVector instead of float variables?	-- Answered in questions doc --
38. Use the PVector class	Used for elements drawn on screen
39. Do some basic physics: use position, velocity, and acceleration (due to gravity) vectors	Don't really have a use for gravity, but since I need it I will add particle effects or smth that fall off screen
40. Find the direction and distance between two points	Useful for measuring distance to enemies
41. Create a random 2D vector	Particle effects
42. What is a normalized vector, why is it useful?	-- Answered in questions doc --
43. Using the Processing documentation look up a method in the PVector class that's new to you and use it in your code.	The only method I haven't used yet is angleBetween()… Guess I'll find a place for it?

				Edit: Also limit() and setMag(), ezpz I can use those no problem. Edit 2: Where would I use those… Edit 3: Start speed for particle effects
Nice to Know (optional)				----------
44. Use a timer				Will be useful to make enemies not move instantly
45. Switch between “game states” (eg grounded/jumping) using conditional statements				Main menu > player turn > enemy turn states
46. Make a button or toggle switch with a roll-over highlight (color or size change)				Menu buttons
47. Create a drag & drop object				Cards? Not sure if I’ll include this one
48. Do animation with images (spritesheet or individual files)				Extension
49. Use collision detection between objects				Making sure players don’t go out of bounds
50. Use the Game Control Plus controller library to get joystick or gamepad input				Going to be used
Milestone 1	Milestone 2	Milestone 3	Milestone 4	
What will I deliver? Multiple players ~~Controller support~~ Placeholder player actions (menu system) Player cards	Rough main menu Player movement Loading content from disk	You are strongly encouraged to deliver your finished game at Milestone 3. Map generation (hard-coded) Gameplay (item collection and use, actions, entities, game end) Particles	EXTENSIONS: No promises! These would be nice to have, but are a stretch and likely won’t be added. Controller support Sprites/Graphics Juicy animations Procedural level generation Enemy pathfinding More cards Sound FAQ	

Which inventory skills will this demonstrate? List them.				
~~50~~ not anymore	49	43	Potential!	Question Numbers:
39 (ish)	45	41	50	42
38		40	48	37
36		33	46	32
35		18 (locking doors)		31
34		17 (locking doors)		27
30				26
29				25
28				22
24				19
23				
21				
20				
16 to 1				
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.**		

Skills Inventory

Outline

Brief