**Pseudocode for menu**

Create variable to store name input (public)

Create variable to store the users mark in the course (double, public static, start with 100%)

Create variable to set a delay time in milliseconds, set to 60.

Write text to title label and “enter your name” label letter by letter (still working on this code)

If the textbox for the name is black, it will disable the start game button. If not, it will enable the start game button.

The help menu shows the controls for the game and how to play when the help button is pressed.

Show the game form when the “start game” button is pressed, hide menu form

**Pseudocode for game (classroom)**

Display message from Mr. Krnic, explaining what the class is to do today.

Get keyboard input from user (arrow keys/wasd , space bar/enter)

If it’s an arrow key or wasd input, turn the character to face that direction.

Afterwards, check if it is possible for the player to move in that direction. If their way isn’t blocked by other objects, move the on-screen character one space over on the grid in that direction, else don’t move them at all.

If its space or enter, initiate interaction with adjacent object(computer, classmate, etc.) If there is no nearby object, do nothing.

**Pseudocode for game (using computer)**

Create a boolean array **(static)** with around 30 elements to represent each of the possible skills that can be learned.(This number will have to be confirmed, not sure exactly how many lessons we had).

User chooses what they want to work on from the Sebnic menu.

If they chose a lesson, show the lesson form and hide the classroom form.

If they chose a practical test, load the practical form and hide the classroom form.

If they have chosen to work on their current project (option not always available), load the project form and hide the classroom form.

(There might be more options from this menu, i.e. play games, etc.)

**Pseudocode for lesson**

Create a local variable to store how well the user does on the lesson.

Display the first question to the user, related to that day’s lesson. Also display the possible answers (multiple choice, true or false) or a rich text box (complete the code, fill in the blank, etc.).

User presses a button to check their answer, and if their answer is correct add one to the lesson-score variable created above.

Repeat, the user will usually be given 5 questions per lesson.

After 5 questions, check their score. If they have gotten over half of the questions right, tell the user they have “learned” that skill. Set the element in the array for that particular skill to true, this represents the fact that the user has learnt it. (skills are numbered in chronological order, so the first skill (unit 1 lesson 1) will be element 0.)

Show the user how well they did on the lesson

Close the lesson form, return them to the computer screen.

**Pseudocode for practical test**

Create a variable to use as a timer (there will be a time limit!)

-This will be quite similar to the lesson code, but it will be all “fill in the remaining code”-type questions and will be much longer, with no multiple choice. The skills that the user has or has not learned so far will affect how well they do in the practical, but how this will affect the questions we have not yet decided.

When the user has completed their practical, re-calculate their course mark and display it.

**Pseudocode for test**