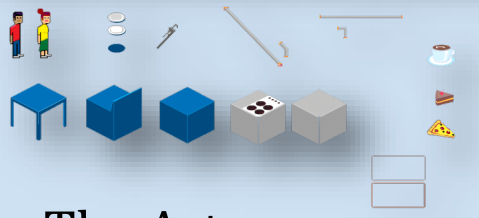
if memory serves

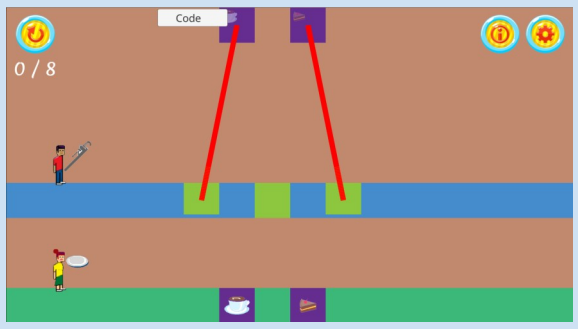
A game to “point” you in the right direction

James Hellman & Warwick New



The Art

Here is an example of the art present in the game, for now only a placeholder until an artist can update the content.



Example Level

Here is a level in which Addy and Val must create the correct pointers to allow the coffee and cake reach the correct tables.

First Addy must create a pointer to the coffee table and Val must pick up the coffee and send it on, in this example only one pointer is allowed at a time so once the coffee has reached the table the pointer must be destroyed and a new one must be created for the cake table.

Open Source

This project is an open source project working with the Godot engine and source controlled by Github.

<u>The project can be found here:</u>

https://github.com/James120393/comp330_port

<u>The engine is available for download here:</u>

https://godotengine.org/

<u>Sources:</u>

[1] Monica M. McGill et al. 2017. If Memory Serves: Towards Designing and Evaluating a Game for Teaching Pointers to Undergraduate Students. In *Proceedings of the 2017 ITiCSE Working Group Reports* (ITiCSE '17). ACM, New York, NY, USA. DOI: https://doi.org/10.1145/3059009.3059037

[2] Chris Johnson et al. 2016. Game Development for Computer Science Education. In *Proceedings of the 2016 ITiCSE Working Group Reports* (ITiCSE '16). ACM, New York, NY, USA, 23-44. DOI: https://doi.org/10.1145/3024906.3024908

[3] Godot Engine Documentation, URL: http://docs.godotengine.org/en/stable/ [Online] Accessed - 08/12/2017

[4] Godot Engine, URL: https://godotengine.org/ [Online] Accessed - 08/12/2017, Paragraph One

If memory serves is an open source game created by a group of researchers that attended the ACM Annual Conference of Innovation and Technology in Computer Science Education (ITiCSE 2017). The game revolves around teaching undergraduates pointers and was designed using a learning outcome based approach. This allowed for a game that facilitated educational games development within computer science.

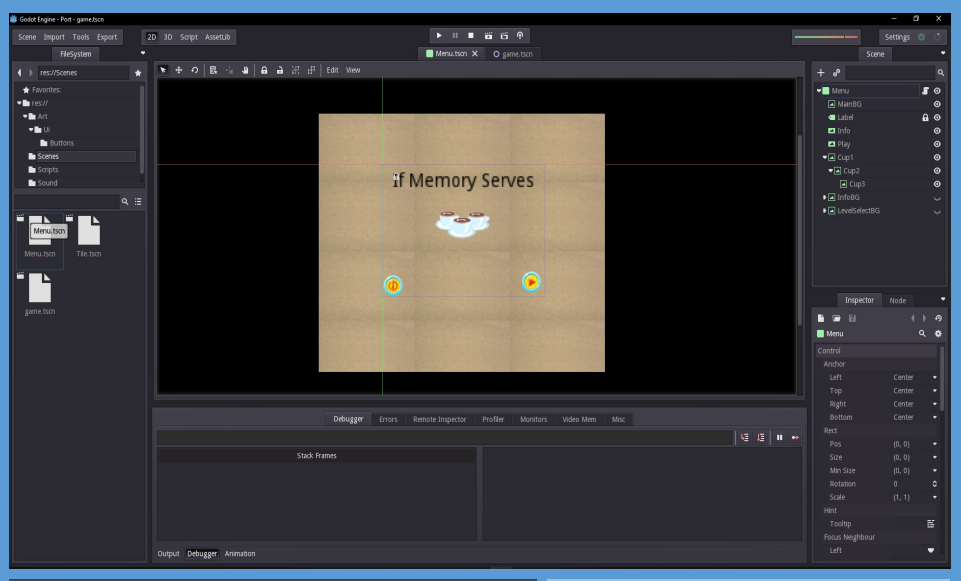
The project proposed in this poster is to port “If memory serves” to an open source games engine (we have chosen Godot) as it is currently being developed on Unity. The challenge here is learning a new games engine and scripting language along with it. Also, the art assets will need slight adjustments to better fit the new engine.

Text file for level setup

```
; Assign a value to a variable in the stack.

; Level Settings
20 ; width
9 ; height
10 ; camera x
4.5 ; camera y
Addy needs to move one cup of coffee from the preparation area to Val at the counter. ; instructions
A ; console target

; Initial Layout
--                  --
--                  --
--                  --
--                  --
-- &                --
--+++++x++++++++++--
--                  --
-- *                --
-------A------------
0 ; number of links

; Target Layout
--                  --
--                  --
--                  --
--                  --
--                  --
--+++++A++++++++++--
--                  --
--                  --
-------?------------
0 ; number of links

; Player Settings
addy valueTool ; addy's available tools
val valueTool ; val's available tools
par 2 ; maximum number of actions required for full reward

; Solution Code
begin
val = 'A';
a = val;
end
```

```
; Initial Layout
--                  --
--                  --
--                  --
--                  --
--                  --
-- &                --
--+++++x++++++++++--
--                  --
-- *                --
-------A------------
0 ; number of links
```

The game's architecture for the level building is constructed around text files. First setting up the dimensions of the level, then instructions and finally, taking in a series of symbols and replacing them with the appropriate in-game objects. The game will wait for the target layout to be achieved and move on once it has been.




```
extends Panel

# member variables here, example:

# var a=2
# var b="textvar"

func _ready():
    set_process_input(true)

func _input(event):
    if get_node("InfoBG/Cross").is_pressed():
        _on_cross_Button_pressed()
    if get_node("LevelSelectBG/Cross").is_pressed():
        _on_cross_Button_pressed()
    if get_node("Info").is_pressed():
        _on_info_Button_pressed()
    if get_node("Play").is_pressed():
        _on_play_Button_pressed()
    if get_node("LevelSelectBG/ScrollContainer/Control/HBoxContainer/Level1").is_pressed():
        _on_Level1_Buttone_pressed()

func _on_info_Button_pressed():
    get_node("InfoBG").set_hidden(false)
    get_node("Play").set_disabled(true)

func _on_cross_Button_pressed():
    get_node("InfoBG").set_hidden(true)
    get_node("LevelSelectBG").set_hidden(true)
    get_node("Play").set_disabled(false)

func _on_play_Button_pressed():
    get_node("LevelSelectBG").set_hidden(false)

func _on_Level1_Buttone_pressed():
    get_tree().change_scene("res://Scenes/game.tscn")
```

Godot uses its own scripting language called GDScript. GDScript is a high level, dynamically typed programming language used to create content. It uses a syntax similar to Python (blocks are indent-based and many keywords are similar) [3].

The Godot engine is a free open source software that can be used to create any variety of game. As quoted on their website “Godot provides a huge set of common tools, so you can just focus on making your game without reinventing the wheel”[4].

This means that it provides all the tools you need to just jump straight into development, without having to program parts of the engine to achieve what your goal.