



EECS 498 : Game Engine Architecture

Winter 2025 Mid-Semester Exam - Answer Packet

By Austin Yarger - University of Michigan (ayarger@umich.edu)

Topic	Score
Game Engine Landscape, History & Misc	/ 40
C++ and IDE Pragmatics	/ 20
Lua and Composition	/ 20
Engine Architecture and Lifecycle Functions	/ 20
Total	/ 100
Total (canvas normalized)	/ 250



Yeahhhh! This is it! Stay cool, and rock on and stuff.
It's ez so don't even worry haha.

Ok then. Let's see what you're made of.
And please ignore May- She gave up on page 2.



Good evening. Before we begin, please repeat after me.
"I have neither given nor received unauthorized aid on this examination, nor have I concealed any violations of the honor code."



name / signature

username

date



Game Engine Landscape, History & Misc

1

The architecture of our course engine (your engine at the conclusion of homework9) is accurately described as an "ECS" **(circle one below)**

___/1

Yes No

2

In designing their Playstation operating system, Sony decided to utilize the open-source Free Berkeley Software Distribution, rather than the more-popular Linux operating system. What is the best rationale for why? **(circle one below)**

___/1

Avoid publishing changes Avoid Licensing Fees Sony created FreeBSD Personnel Fit

3

While Data-Oriented Programming has increased in popularity in certain fields, Object-Oriented programming remains dominant. Why? **(circle one below)**

___/1

Cache Utilization Reduced dev time / cost Security Requirements Licensing Parallelism

4

This course's engine API most closely resembles that of **(circle one below)**

___/1

Jolt Unreal Godot Unity Flash

5

Today's integrated engines (such as Unity, Unreal, etc) are wildly popular, but some prefer to use other categories of engine in search of- **(circle two)**

___/1

Improved Documentation Beginner Friendliness Minimalism Feature-Fullness Community Support

6

It is often perilous to alter a container while it is being iterated through. What are two ways to deal with this problem? **(circle two)**

___/1

Collect-then-alter Alter-then collect Try-then-catch Backwards iteration.

7

The 2D affine transformation matrix is a ____ matrix (rows x columns) **(circle one below)**

___/1

1x1 2x2 3x3 3x4 2

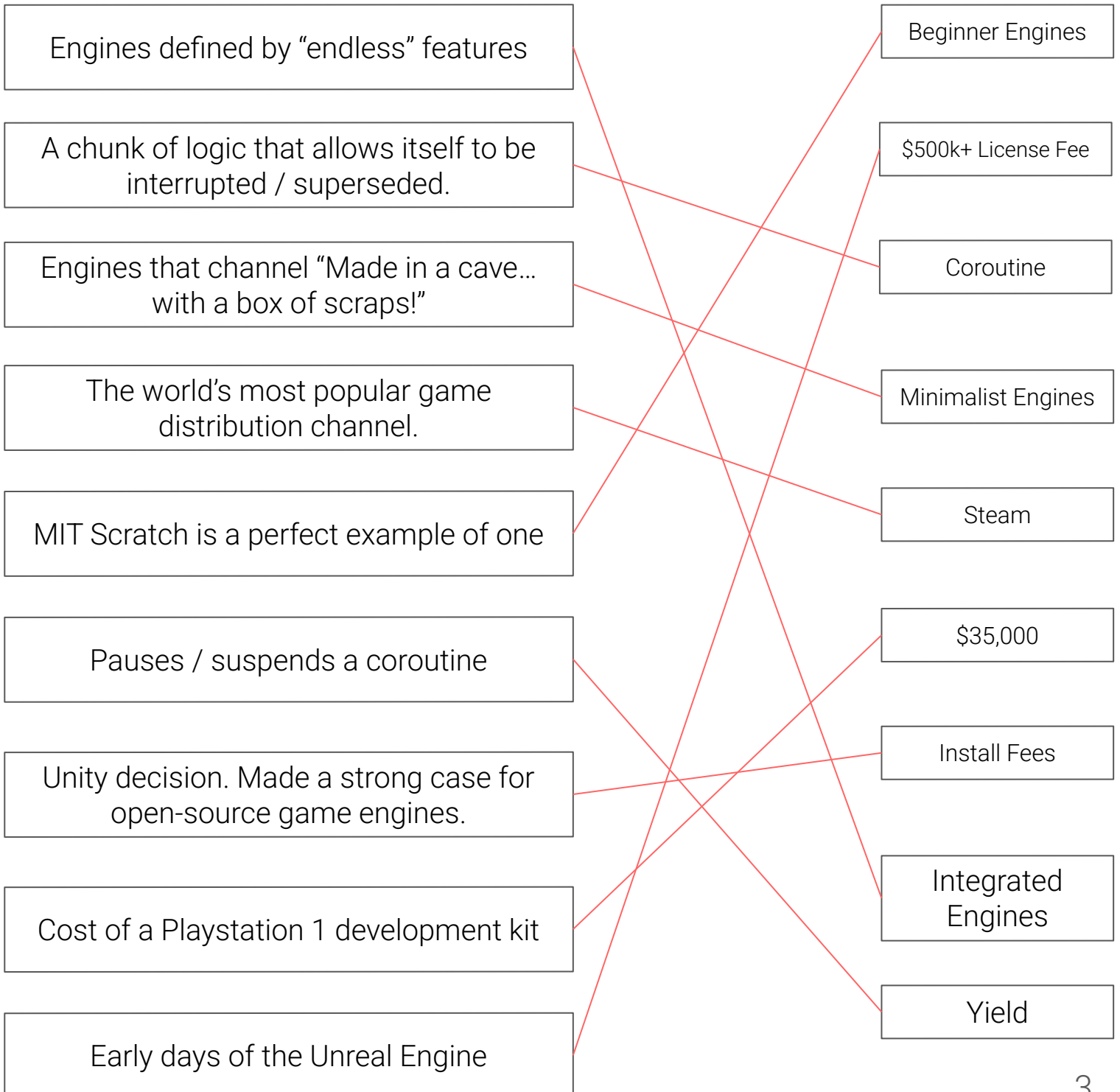


Game Engine Landscape, History & Misc

8

Match each term below with its single most-reasonable definition at the bottom of the page. Do so by drawing a line between two boxes. Each box may only be connected to one line. There will be nine lines total.

___/9





Game Engine Landscape, History & Misc

9

___/8

Match each term below with its single most-reasonable definition at the bottom of the page. Do so by drawing a line between two boxes. Each box may only be connected to one line. There will be eight lines total.

AABB

Rotate my character

Analogous to a "collider"

Do nothing. Win.

Powered by b2ContactListener

Translate and scale my character

Bullet

Havok

1	0	0
0	1	0
0	0	1

b2Fixture

2	0	2
0	2	3
0	0	1

Open Source

0	-1	0
1	0	0
0	0	1

Algorithm

OnTriggerEnter

Commercial

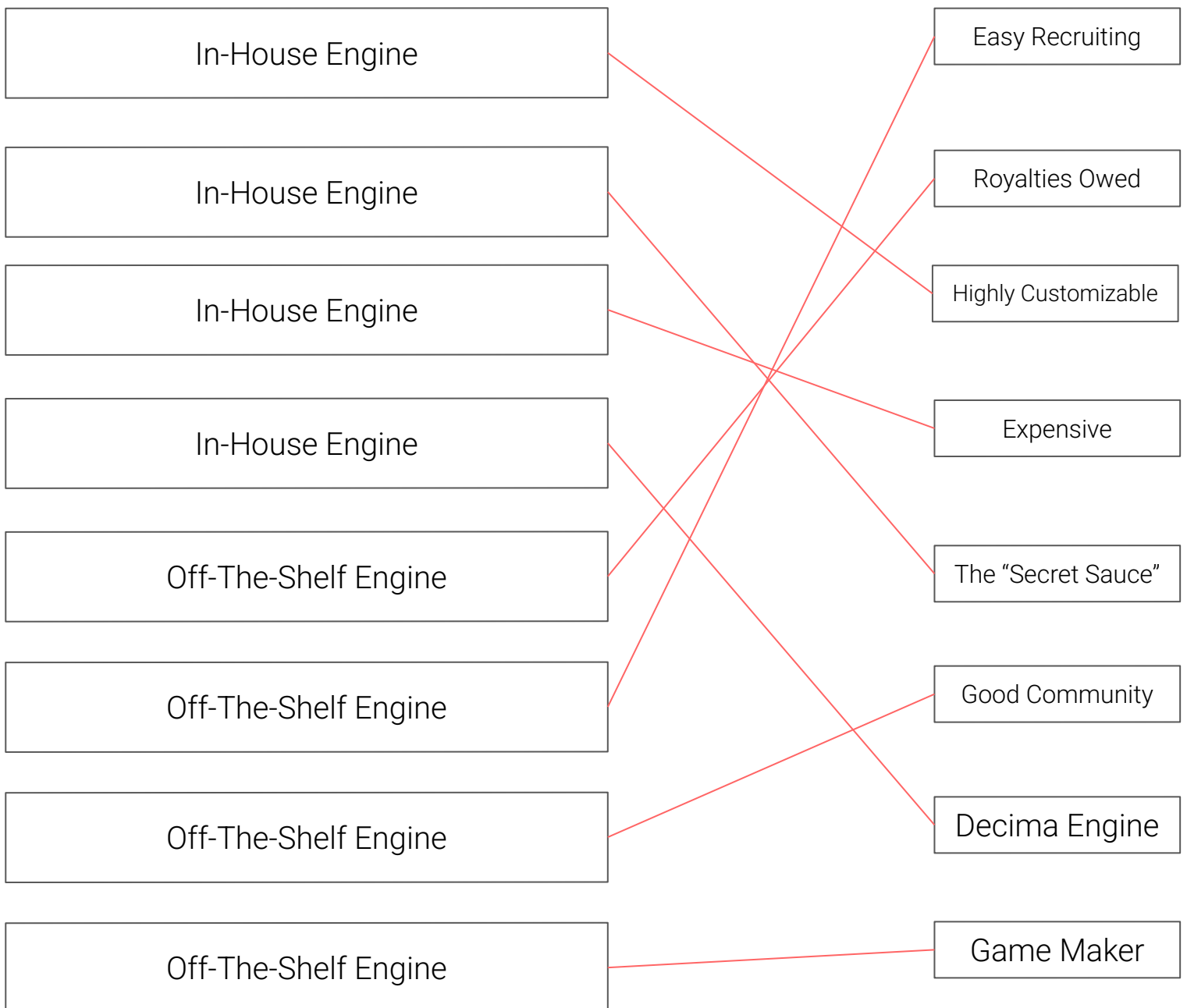


Game Engine Landscape, History & Misc

10

___/8

Match each term below with its single most-reasonable definition at the bottom of the page. Do so by drawing a line between two boxes. Each box may only be connected to one line. There will be eight lines total.



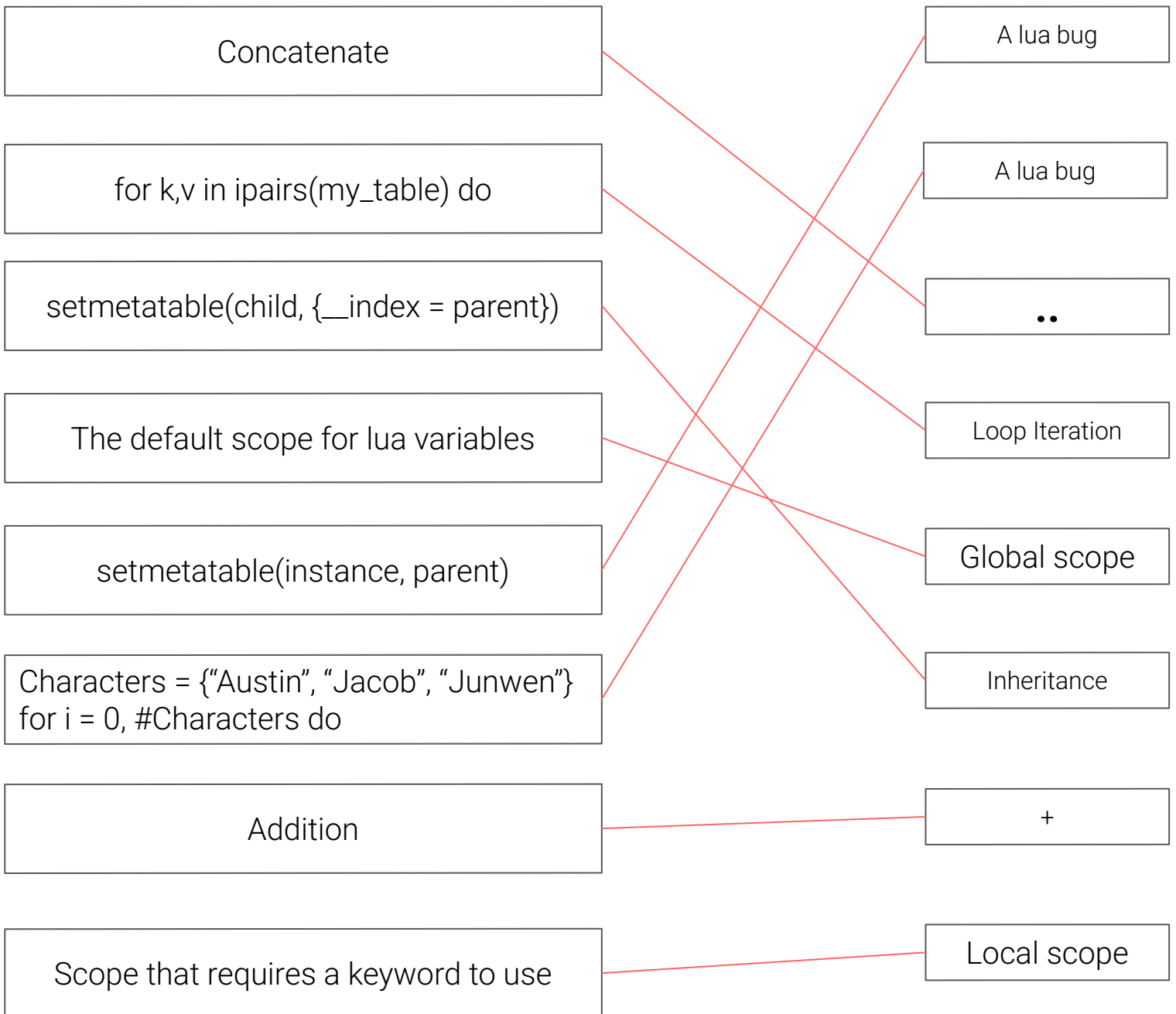


Game Engine Landscape, History & Misc

11

___/8

Match each term below with its single most-reasonable definition at the bottom of the page. Do so by drawing a line between two boxes. Each box may only be connected to one line. There will be eight lines total.



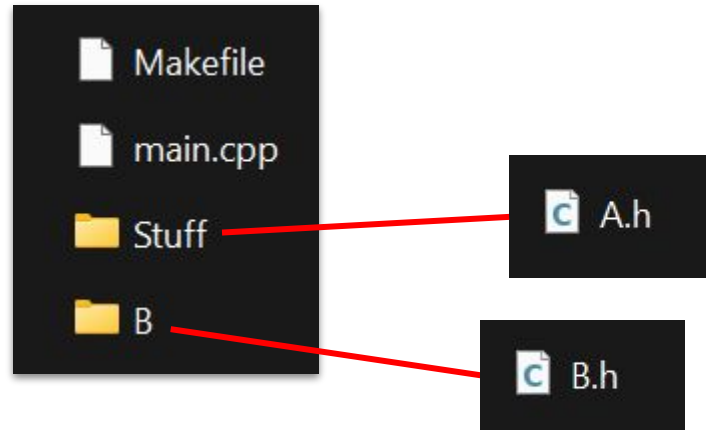


C++ and IDE Pragmatics

```

1  #include <iostream>
2  #include "A.h"
3  #include "B/B.h"
4
5  int main()
6  {
7      // Code that uses
8      // all libraries here
9      return 0;
10 }
11

```



A game engine project that utilizes two non-STL libraries (A and B)
 Library A is header-only. Library B is header-and-binary
 (lib B is installed globally with a binary called "B").

12

___/3

Consider only the Linux operating system / clang for the following questions. Please distinguish between upper-case I and the lower-case l

Write the include / header flag(s) needed to compile the above script.
 (ie, what you would put into a Makefile)

Note : Assume the preprocessor does not include root directory by default.

`-I./ -I./Stuff`

13

___/3

Write the library flag(s) needed to compile the above script.
 (ie, what you would put into a Makefile)

`-lB`

14

___/3

Write a full Makefile (housed in the root) that compiles the code with clang++, has O3 optimization, and has an output binary named "game_engine_lin"

`clang++ ./*.cpp -I./ -I./Stuff -lB -O3 -o game_engine_lin`



C++ and IDE Pragmatics

15

___/2

A friend has implemented an expensive collision-detection algorithm for their chess-like, grid-based game, but is now having performance problems.

For their gameplay mechanics, they only need to know if any Actor* exists at a given cell position (x,y) or if it is empty. Provide a reasonable data structure that answers in O(1) time.

```
std::unordered_map<int, std::unordered_map<int, Actor*>>
```

Briefly explain how it can be used to answer the query.

Use x value of cell to index into first map, then y value of cell to index into second map. If Actor* is nullptr, no one is at this cell. Otherwise, the Actor pointed to by Actor* is there.

16

___/9

While developing a survival / scavenging / crafting game, you begin to encounter performance issues. Your colleague (from MSU) is a fine programmer, but you believe performance may be improved.

```
1 struct Character {
2
3     std::string name;
4     char type;
5     int score;
6     int hunger;
7     int health;
8     int lives;
9     Image* sprite;
10    Inventory* inventory;
11};
```

Character objects for an arcade-style survival / scavenging game.

```
1 health -= hunger
2 if (health <= 0)
3     lives -= 1;
```

Game design pseudocode describing how hunger impacts health and how health impacts life count every frame of the game.



C++ and IDE Pragmatics

Approximately how large is the full Character struct in bytes? Ignore optimizations and issues such as alignment (**hint : use the glossary**).

57

If we were to ignore / strip out data members that are irrelevant to the previous game design pseudocode, what size (in bytes) does our struct become?

12

If we compute with this stripped-down object instead of the full Character struct, roughly how many more objects can we fit in cache? (2x more? 3x more?) (very rough is fine)

~5x more

Write a very cache-friendly implementation of the game design pseudocode

```
// Add your new / altered data structure(s) here.
// Assume they get "automatically" filled with data.

int hunger[];
int health[];
int lives[];

int num_chars; // "automatically" gets set to the number of characters.

int main()
{
    // Perform the computation for all characters here.

    for (int i = 0; i < num_chars; i++)
    {
        health[i] -= hunger[i];
        if (health[i] <= 0)
            lives[i] --;
    }

    return 0;
}
```



17

___/10

resources/component_types/CutsceneManager.lua

```
CutsceneManager = {  
  
    slide = 1,  
  
    IsGameOver = function(self)  
        local may_lives = Actor.Find("May"):GetComponent("Status").lives  
        local magna_lives = Actor.Find("Magna"):GetComponent("Status").lives  
        return may_lives <= 0 and magna_lives <= 0  
    end,  
  
    IsVictory = function(self)  
        local enemies = Actor.FindAll("enemy")  
        return #enemies <= 0  
    end,  
  
    OnUpdate = function(self)  
  
        if self.IsGameOver() then  
  
            Image.DrawUI("gameover", 0, 0)  
  
        elseif self.IsVictory() then  
  
            Image.DrawUI("victory", 0, 0)  
  
        elseif self.slide < 4 then  
  
            Image.DrawUI("teamup" .. self.slide, 0, 0)  
  
        end  
  
        -- Advance slides on spacebar press  
        if Input.GetKeyDown("space") then  
            self.slide = self.slide + 1  
        end  
  
    end  
  
end
```

}



```
CharacterSelectManager = {

    spawned_chars = {},
    hovered_index = 1,

    OnStart = function(self)

        -- Read the global data table.
        for i,c in ipairs(character_data) do

            -- Spawn characters from template
            local char_actor = Actor.Instantiate("Character")

            -- Initialize char data.
            local char = char_actor:GetComponent("Character")
            char.x = i*100
            char.y = 300
            char.data = c
            table.insert(self.spawned_chars, char)
        end
    end,

    OnUpdate = function(self)

        -- Character select controls
        if Input.GetKeyDown("right") then
            self.hovered_index = self.hovered_index + 1
        end

        if Input.GetKeyDown("left") then
            self.hovered_index = self.hovered_index - 1
        end

        -- Wrap the hovered_index so we don't go out-of-bounds.
        if self.hovered_index > #spawned_chars then
            self.hovered_index = 1
        elseif self.hovered_index < 1 then
            self.hovered_index = #spawned_chars
        end

        -- Let the Character component know we've chosen them.
        if Input.GetKeyDown("space") then
            self.spawned_chars[hovered_index]:Select()
        end
    end
end

}
```



Lua and Composition

resources/component_types/Character.lua

```
Character = {

    OnStart = function(self)
        self.anim_frames = -1
    end,

    OnUpdate = function(self)
        if self.anim_frames > 0 then

            -- Our character has been chosen. Render cool pose.
            self.anim_frames = self.anim_frames - 1
            Image.DrawUI(self.data.pose, self.x, self.y)

        elseif self.anim_frames == 0 then

            -- Done posing, so let's begin gameplay.
            Scene.Load("gameplay")

        else

            -- Render normal idle pose while we wait to (hopefully) be chosen.
            Image.DrawUI(self.data.idle, self.x, self.y)

        end
    end,

    Select = function(self)

        -- The manager (via the player) has chosen us!
        -- Play vocal clip and begin our cool pose.
        self.anim_frames = 120
        Audio.Play(1, self.data.clip, false)

    end

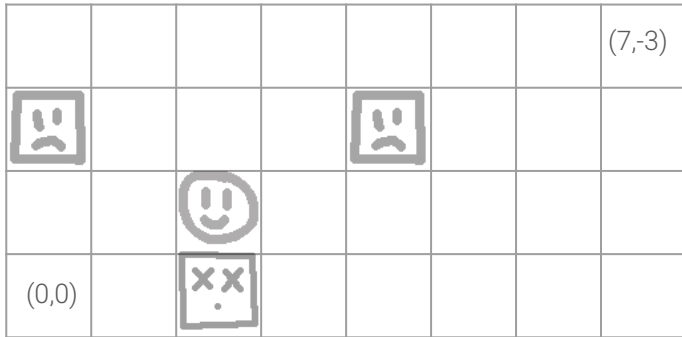
}
```



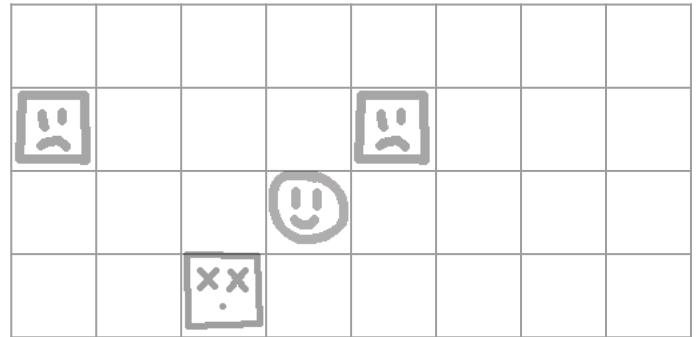
Engine Architecture and Lifecycle Functions

19

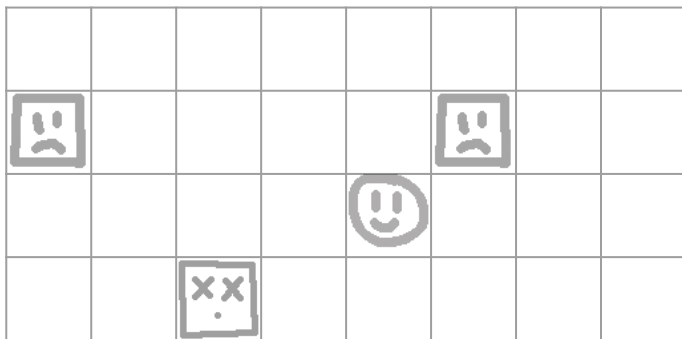
__/9



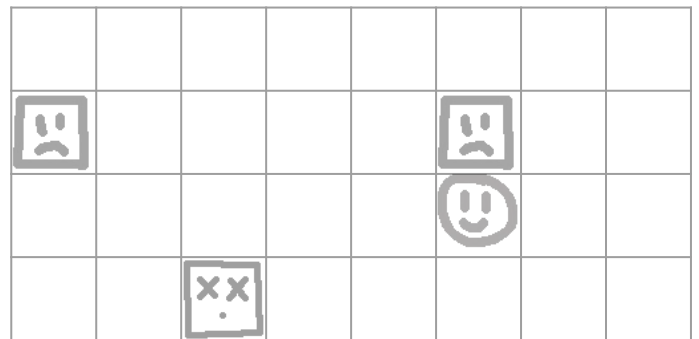
Frame #2 (input coming early frame : **up**)



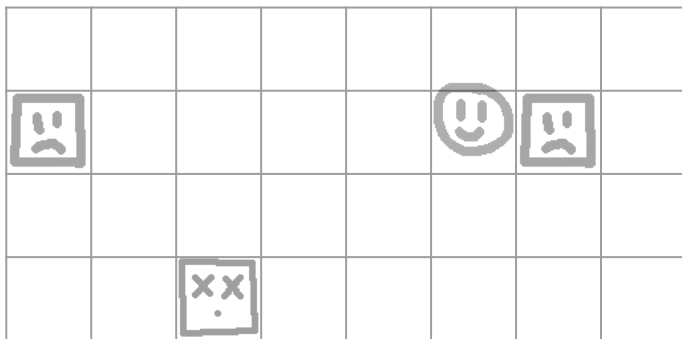
Frame #3 (input coming early frame : **right**)



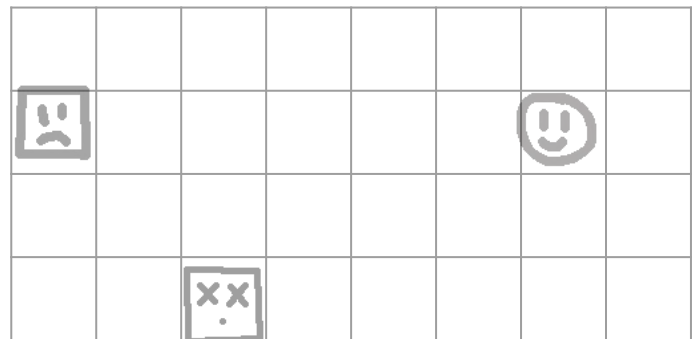
Frame #4 (input coming early frame : **right**)



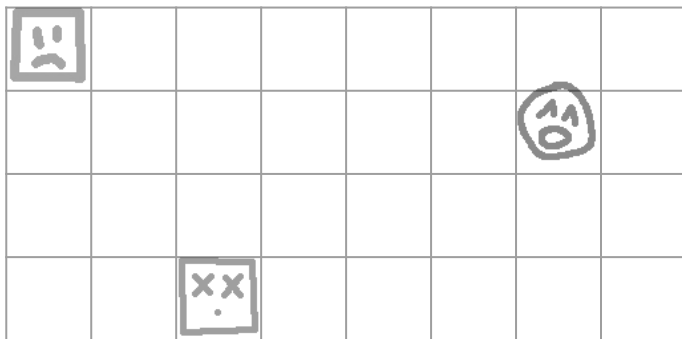
Frame #5 (input coming early frame : **right**)



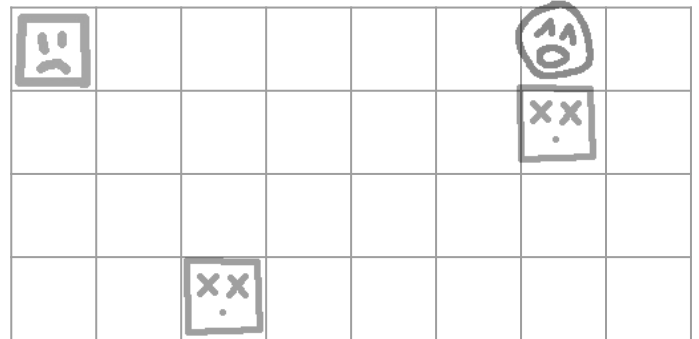
Frame #6 (input coming early frame : **up**)



Frame #7 (input coming early frame : **right**)



Frame #8 (input coming early frame : **none**)



Frame #9 (input coming early frame : **up**) 13



20

__/10

```
class Actor
{
public:
    bool active;
    void OnStart();
    void OnDestroy();
    LuaRef GetComponentByKey(const string & key);
    static Actor* GetActorByID(int actor_id);

    // A c++ function that may be called from lua.
    // It "registers" a provided lua function (with a particular component key)
    // to receive network data should any network data arrive for it.
    void Connect(string comp_key, LuaRef receive_function); // TODO : implement.

    // A c++ function that may be called from lua.
    // It "unregisters" all network functions associated with this actor.
    void Disconnect(); // TODO : implement.

    // Called automatically by the engine when network data arrives.
    // Funnels data to any registered network functions on any components.
    void NetworkDataArrived(const NetMessage & message); // TODO : implement

    // TODO : Add one data structure below. Efficient choice = full points.

    std::unordered_map<std::string, std::vector<LuaRef>> net_functions.

};
```



Engine Architecture and Lifecycle Functions

game_engine_uniqname/src/Actor.cpp

```
// This may be called from lua. It has been injected into the actor LuaRef and
void Actor::Connect(string comp_key, LuaRef receive_function)
{

    net_functions[comp_key].push_back(receive_function)

}
```

game_engine_uniqname/src/Actor.cpp

```
// This may be called from lua. LuaBridge has been properly configured.
void Actor::Disconnect()
{

    net_functions.clear();

}
```



Engine Architecture and Lifecycle Functions

```
// A data structure that represents a message from the server / other clients.
class NetMessage
{
public:
    int actor_id; // Use these two fields to reach relevant lua components.
    std::string component_key;
    std::string data; // Deliver this to lua functions.
};
```

game_engine_uniqname/src/Actor.cpp

```
// This function gets called automatically when net data arrives for this actor.
// Reminder : Calling a lua function on a component requires a "self-reference"
// of // the component to be "sent-in" (check Glossary for example).
// NOTE : Do not call the lua function if the actor isn't active
// or if the component isn't enabled.

void Network::OnNetworkDataArrived(const NetMessage & message)
{

    // Get the relevant actor and component.
    Actor* actor = Actor::GetActorByID(message.actor_id);
    if (actor->active == false)
        return;

    LuaRef component = actor->GetComponentByKey(message.component_key)
    if (component["enabled"].cast<bool>() == false)
        return;

    // Iterate through the registered functions.
    for (LuaRef & function : net_functions[message.component_key])
    {
        // Functions in lua must be called with self reference as first param.
        // IE, a reference to the component.
        function(component, message.data);
    }

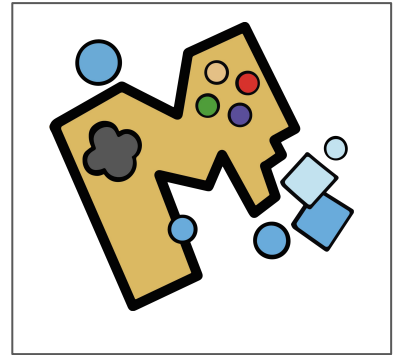
}
```




Engine Architecture and Lifecycle Functions

21

Name your new engine and draw its logo ->
(you need not keep the name "A2 Engine")



__/1

A2 Engine

Thanks for
playing



Prepared by Maylen Meguri, (Donna, May, and Mags' artist) specifically for the students of 498

