



EECS 498 : Game Engine Architecture

W26 Mid-Semester Exam - Answer Packet

Topic	Score
Game Engine Landscape, History & Misc	/ 30
C++ and IDE Pragmatics	/ 20
Lua and Composition	/ 20
Engine Architecture and Lifecycle Functions	/ 20
Total	/ 90
Total (canvas normalized)	/ 250

Alright agent– **focus**. You’ve totally got this. No big!
This mission is all you... but I’ll lend a hand when I can.



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

___/1

Off-the-shelf game engines were a rarity in the 80s and 90s, forcing most development teams to build their own engines from scratch. Off-the-shelf engines still existed in this era however. Which of the following was widely available (and accessible without fees / paid licenses) first? **(circle one)**

Unreal Engine Godot Unity UbiArt Game Maker

2

___/1

In 2026, two off-the-shelf game engines form a plainly-visible duopoly, consuming much of the game development and visualization markets. Which open-source engine is closest to disrupting them? **(circle one)**

Game Maker Unreal PyGame Godot Unity

3

___/1

The 3D affine transformation matrix is a _____ matrix (rows x columns) **(circle one below)**

2x2 3x3 4x4 5x5

4

___/1

What does the acronym "ECS" stand for?

5

___/1

How does "ECS" differ from the architecture adopted by our engines throughout homeworks 7 and 8? (briefly mention one substantial difference)

6

___/1

How may we temporarily pause execution of a lua coroutine? **(circle one)**

Pause Suspect Yield Delay NoOp



Game Engine Landscape, History & Misc

7

___/8

The 2D affine transformation matrix, along with its 3D sibling, are some of the most fundamental matrices in all of computer graphics and game development, providing an efficient representation of common movement, rotation, and scale operations. It is a mainstay of popular engines such as Unity, Unreal, and Godot.

Write the 2D affine transformation matrix in the cells to the right, supporting translation (T_x , T_y), Rotation (θ), and scale (S_x , S_y). Fill every empty cell with something.



Psssst!
Top-left is $S_x \cos(\theta)$.
Center is $S_y \cos(\theta)$.

Use the completed 2D affine transformation matrix above to create a transformation matrix that will translate actors by an (x,y) of $(2, -1)$ while also scaling by a factor of positive 3 in both dimensions.

T =

Use the transformation matrix "T" above to transform the following three position vectors.

$$T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \\ \\ \end{bmatrix}$$

$$T \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} \\ \\ \end{bmatrix}$$

$$T \begin{bmatrix} -2 \\ -3 \\ 1 \end{bmatrix} = \begin{bmatrix} \\ \\ \end{bmatrix}$$



Game Engine Landscape, History & Misc

8

___/8

In pseudocode, write the AABB algorithm in the box below. Your algorithm is supplied with an `SDL_Rect a`, and a second `SDL_Rect b`. `SDL_Rect` has `{x, y, w, h}`. If `a` and `b` share a corner or border, this is **not considered colliding**.

Utilize the AABB algorithm to determine which of the following rectangles are colliding with which other rectangles. If two rectangles are colliding, draw a line between them. All rectangles are defined in `SDL_Rect` format `{x,y,w,h}` and the coordinate system is that of SDL. Multiple rectangles that share borders or corners are not considered to be colliding. The anchor / pivot point for each rectangle is its top-left corner.

`{0, 0, 5, 4}`

`{5, 0, 1, 1}`

`{0, 0, 1, 1}`

`{5, -1, 1, 1}`

`{-5, 0, 1, 1}`

`{4, -2, 2, 10}`



A tip : draw it out!



Game Engine Landscape, History & Misc

9

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.

___/8

In-House Engine

Zenilib

In-House Engine

Adobe Flash

In-House Engine



Retro Engine

In-House Engine

Crystal Tools

Off-The-Shelf Engine

Clickteam Fusion

Off-The-Shelf Engine

Frostbite

Off-The-Shelf Engine

Stepmania

Off-The-Shelf Engine

UbiArt



C++ and IDE Pragmatics

10

___/8

Order the following operations from lowest latency (1) to highest latency (8) according to course lectures, demonstrations, and homeworks. Write the number / ranking in the empty boxes to the right of the operation.

SSD random read

L2 cache reference

Deliver letter via United States Postal Service

L1 cache reference

Local network round trip

Global network round trip

Deliver letter yourself within city (driving)

Main memory reference



C++ and IDE Pragmatics

11

__/2

`std::shared_ptr` is popular among developers for several reasons, but receives criticism for having slightly-inferior performance characteristics compared to that of a `raw pointer` or `std::unique_ptr`.

In the box below, briefly describe what causes this reduction in performance.

12

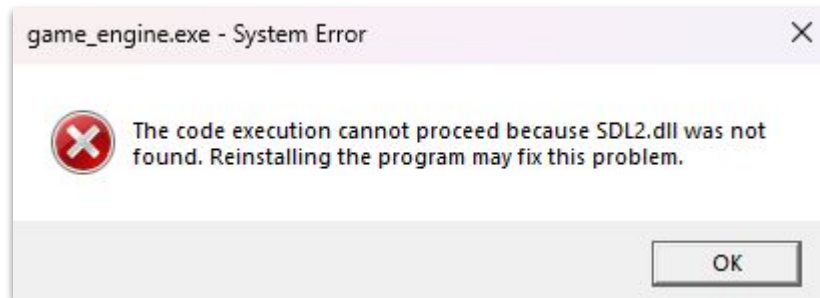
__/4

According to the course, what are the four primary “questions” of c++ compilation? (ie, what four pieces of information must be specified to build).

13

__/2

Having provided your compiler and linker with all of the information necessary to build (the four pieces of information listed above), your build completes and your program finally begins to run. Suddenly, your program crashes with the following error--



Note from AY :
I forgot to make problem #13 ask for something, so everyone got 2 points.
Lucky number 13 lol.

14

__/2

This cannot be a compiler issue, as the compiler’s job has long since finished. Nor can it be the linker’s problem, as the linker successfully created the final binary. Name the technology responsible for this post-build-process error.

There are multiple correct answers





C++ and IDE Pragmatics

15

Examine the following file structure and fix the post-build-process issue seen previously. Move a file by crossing it out and writing its name elsewhere. You need only solve the issue seen in the previous problem.

__/2

project/glm/simd/

- common.h
- exponential.h
- geometric.h
- integer.h
- matrix.h
- neon.h
- packing.h
- platform.h
- trigonometric.h
- vector_relational.h

project/x64

Release

project/

- game_engine
- glm
- rapidjson
- SDL_image
- SDL_mixer
- SDL_ttf
- SDL2
- src
- x64
- game_engine.sln

project/SDL_ttf/lib/

- SDL2_ttf.framework
- SDL2_ttf.dll
- SDL2_ttf.lib

project/SDL2/lib/

- SDL2.framework
- SDL2.dll
- SDL2.lib
- SDL2main.lib
- SDL2test.lib

project/src/

- Actor.cpp
- Actor.h
- ActorTemplateDB.cpp
- ActorTemplateDB.h

project/x64/Release/

- game_engine.exe
- game_engine.pdb



Lua and Composition

16

—/10

resources/component_types/KPManager.lua

```
KPManager = {
```



Lua and Composition

```
-- More space for KPManger
```

```
}
```



Lua and Composition

17

__/10

resources/component_types/GlobalHighScoreManager.lua

```
GlobalHighScoreManager = {
```



Lua and Composition

```
-- More space for GlobalHighScoreManager
```

```
}
```



Engine Architecture and Lifecycle Functions

18

__/10

							(7,-3)
(0,0)							

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

Frame #3 (input coming early frame : **none**)

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

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

Frame #6 (input coming early frame : **none**)

Frame #7 (input coming early frame : **none**)

Frame #8 (input coming early frame : **right**)

Frame #9 (input coming early frame : **space**)



Engine Architecture and Lifecycle Functions

19

__/10

```
class AudioDB
{
public:
    // Connected to Lua functions Audio.RegisterBeat() & Audio.UnregisterBeat()
    static void RegisterBeat(int channel, LuaRef function);
    static void UnregisterBeat(int channel, LuaRef function);

    // Called every frame automatically.
    static void OnUpdate();

    // Obtain the channels currently playing audio?
    static std::vector<int> GetActiveChannels();

    // Obtain information about a particular audio channel.
    static ChannelInfo GetChannelInfo (int channel);

    // Obtain information about a particular audio file.
    static AudioFileInfo GetFileInfo (const std::string & filename);

    // TODO : Write a data structure below to track your function registrations.
    // Tip : LuaRef has equality operator defined, but is NOT hashable.

    // TODO : Write a structure below to track the previous beat number per channel.

};
```

```
struct ChannelInfo
{
    float seconds_since_play = 0; // This increases every frame.
    std::string filename;
};
```

```
struct AudioFileInfo
{
    std::string filename;
    float seconds_per_beat; // time between beats (1.0 / beats-per-second)
};
```



Engine Architecture and Lifecycle Functions

```
class EngineUtils
{
public:
    // Use to get the "self" reference of a function (the component it's on).
    // NOTE : You may assume all registered functions are inside a component.
    // NOTE : You may assume all registered functions are member functions.

    static LuaRef GetCompOfFunction(LuaRef function);
};
```

```
// This may be called from Lua via the Audio table.
// Use data structures added within AudioDB (Reminder : this function is static).
// Assume no function will be registered for the same channel multiple times
void AudioDB::RegisterBeat(int channel, LuaRef function)
{

}
```

```
// This may be called from Lua via the Audio table.
// Use data structures added within AudioDB (Reminder : this function is static).
void AudioDB::UnregisterBeat(int channel, LuaRef function)
{

}

}
```



Engine Architecture and Lifecycle Functions

```
// This function gets called automatically every frame.
// 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 worry about the state of the actor or component (enabled, etc).
// Reminder : This function is static.
void AudioDB::OnUpdate()
{

}

}
```



Engine Architecture and Lifecycle Functions

20

___/1

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

Thanks for
playing



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

