**TEMASEK POLYTECHNIC**

**SCHOOL OF INFORMATICS & IT**

**DIPLOMA IN GAME DESIGN & DEVELOPMENT**

**AY2023/2024 OCTOBER SEMESTER (LEVEL 2) TERM A**

**GAME MATH AND PHYSICS (CGE2C15) TERM A**

**Project Documentation (10%)**

|  |  |
| --- | --- |
| **Class** | P03 |
| **Name as in register** | Low Tong Ee Thom |
| **Group #** | **5** |
| **Topic covered** | Particle system |
| **YouTube URL** | Make sure the link is shareable! |

**You must submit:**

* **A discussion about the** **Technical Problems & Solutions encountered**
* **A reflection about this assignment, and also of GMAPS.**

**Each question is on a separate page below.**

1. **Technical Problems & Solutions**

**Discuss the *technical* problems you faced while writing your code, and how you overcame them.**

**Technical problems do NOT include problems you might have had with time-management, knowledge of basic GMAPS topics, illnesses, part-time work, broken computers, lost files, or other distractions, etc.**

**Focus on**

* **Problems you had in understanding concepts related to the chosen topic, and how you managed to gain a better understanding, e.g., via online references, use of ChatGPT and other AI, etc. Make sure you list all references used.**
* **Problems you had in coding your demo implementation, and how you managed to overcome these. Give specific examples.**

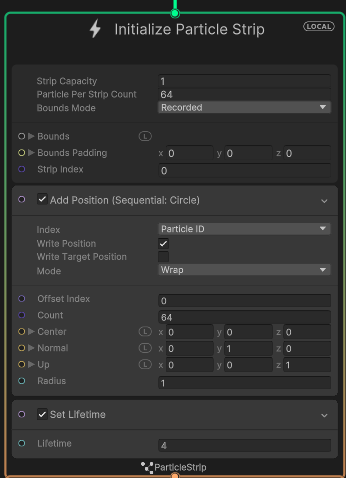
**Introduction:**

As this is my first time working with VFX graph, I find hard to understand on how to even make particles. So, when I first started out, I decided to have a road map to know where to start.

1. Try to create a single strand of Grass.
2. Making a patch of grass
3. Adding additional features.

**Making Grass:**

The most difficult thing was getting the grass to be rendered as a particle. There were only two ways to render the particle. Individual particle or as a particle strip. I understood how to render using individual particle, but I find it hard to render them together using the strip.



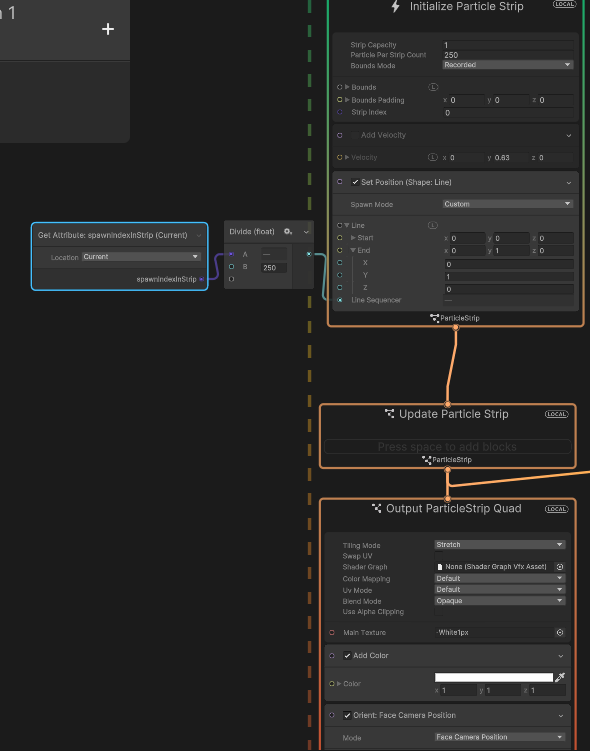
At first, I could not figure if the particle strip position is added to either the particle strip or the individual particles in the strip.

Even after reading the [documentation](https://docs.unity3d.com/Packages/com.unity.visualeffectgraph@12.1/manual/Block-IncrementStripIndexOnStart.html?q=particle%20strip), it still does not give a good idea on how the particle strip works. In fact, there is nothing stated on how to use particle strip on the unity documentation.

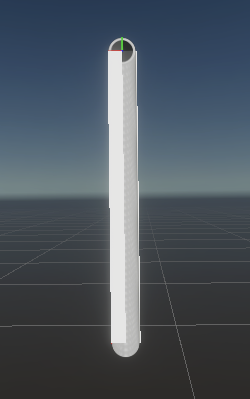
***How I solve it:***

I did quite a bunch of research on how to render particle strip. What help me was the 6-part tutorial video series by [Thomas Iché](https://www.youtube.com/@thomasiche8734/playlists). Where it the 4th and 6th tutorial teach how to use particle strip for visuals like lighting and hair.

When spawning the grass, the particle strip will apply changes to every particle assign to that strip. So, the changes will be applied to every particle in the initialize particle strip. The mesh will then be made as Unity connect the particles together to form a **single quad** which can be used to apply the texture.



How a sample visual script look like to render a line using particle strip



How the strip would look like in Unity

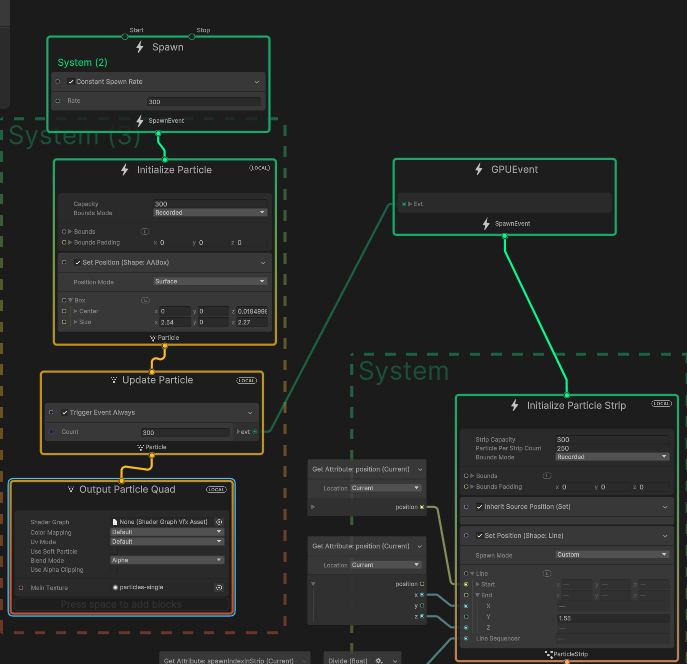
**Figuring out how to render a patch of Grass**

The next difficulty come which is how to render particles in a particle? Throughout the documentation, there were nothing about being able produce more particles from existing particles. So, I need to know,

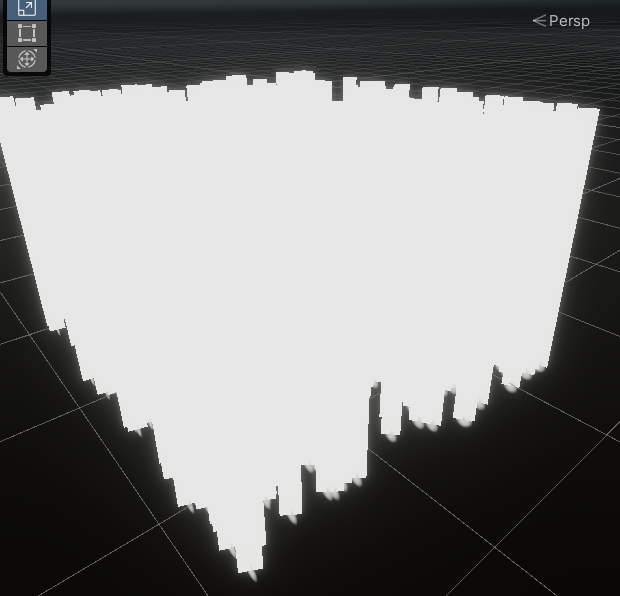
1. How to have particles to throw out more particles (or particle strip)
2. How to group up this particles and particle strip to form a terrain

**How I solve it:**

Thanks to [Thomas Iché](https://www.youtube.com/@thomasiche8734/playlists) , his 6th tutorial introduce GPU events and how to utalise them. It was not stated in the unity documentation since it is considered an **experimental block. GPU events** allow particles to throw out event in the GPU to signal other listeners like an observer pattern. This makes particle call out events to start spawning other particles.



A simple script to make a square patch using particle system



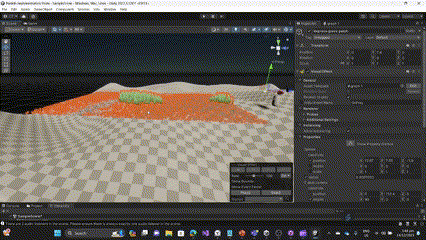
The end result is a square patch using GPU event.

**Rendering the grass in a terrain:**

Next is how the grass should render in a terrain. There weren’t a lot of examples I could reference to learn how to make particles spawn in a randomly procedure terrain. Another problem was that VFX graph do not support spawning attributes that are varying shapes like a terrain. The only spawning attributes VFX graph that does that is through reading a noise map which I can’t use since the terrain and VFX graph use different noise map. This was by far the hardest problem I had when trying to render the grass.

**How I solve it:**

While brainstorming, I realise that, if the particles were to collide with the terrain, it would know where the collision happen in that point of time and would spawn the grass strand. After much research, I get to know about “[collide with depth buffer](https://docs.unity3d.com/Packages/com.unity.visualeffectgraph@10.2/manual/Block-CollideWithDepthBuffer.html)” attribute that VFX graph provides. Where it will stop the particles when it see a visible object in the scene. With this knowledge, I can set the collision to kill the particle the moment it hits the terrain surface. I can the send a event to spawn the particles the moment it dies. That way, the grass will know where to spawn in the terrain and it will randomly generate it by itself.



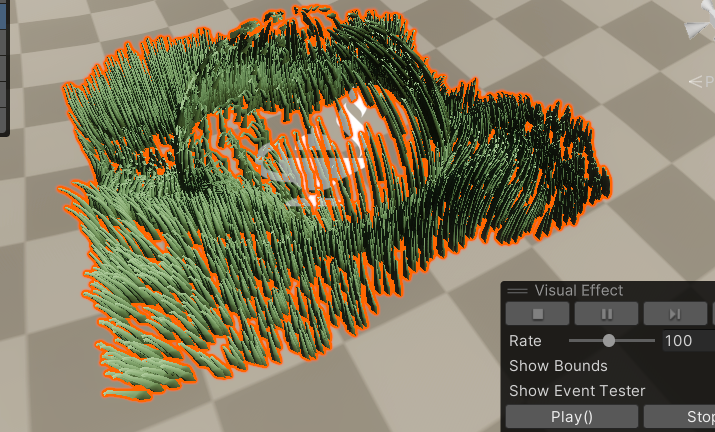
How the grass production works

**Adding (additional stuff) physic interactions:**

I wanted to be more ambitious for this project and I thought it would be cool to have some physic interaction to the grass. The problem was figuring out how to add this physic interaction to the grass patch.

**Adding collision to the grass patch:**

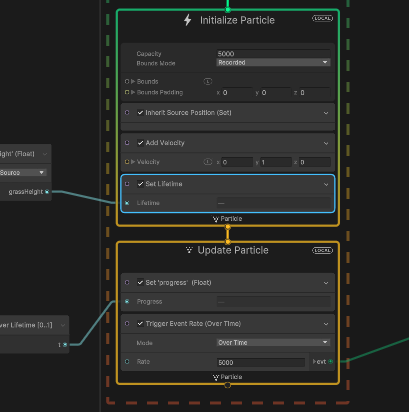
While trying to add collision to the grass, I had a lot of issues with setting it up due to how my grass was set up initially. My grass initially would wrap around the grass because the particles in the grass strand will move further away from the each other as it is rip apart by the sphere collider.



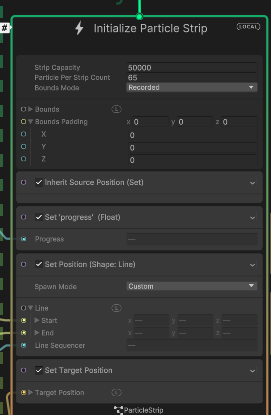
An example of the problem I had initially with the grass

**How I solve it:**

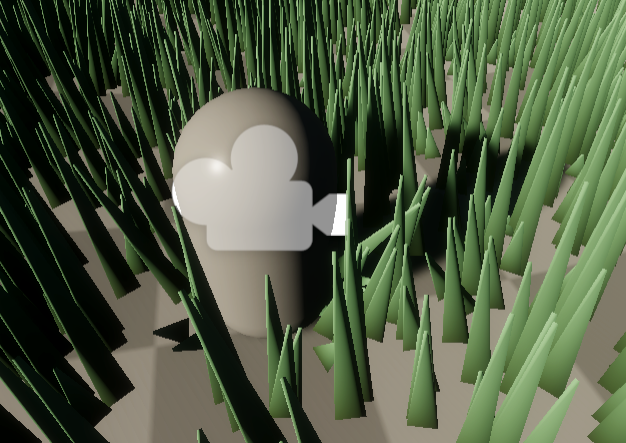
Initially the grass patch strip mesh would be produced due to the velocity of the particles that is shot up. Which cause it to then wrap around the collider as it tries to avoid hitting it. After much trial and error, I manage to make the grass strip mesh to be produce based of that line, this prevents the grass from wrapping around the collider as a result.



Before improvement (where the grass patch would wrap around the collider)



After improvement (where it would maintain structure no matter what)



Now the grass would not wrap around the sphere collider

1. **Reflections**

**Reflect on:**

* **Your project, including the presentation and implementation parts. Discuss how useful you found the project work (e.g., relevance to game development, development of your technical knowledge, development of your programming skills, etc.) and how you approached it (e.g., your attitude, self-discipline, independence in learning, etc.).**
* **GMAPS as a whole. You can also use the discussion points suggested above here (e.g., relevance to game development, your attitude, etc.).**

1. **Project Reflection**

**Before starting on the project:**

Initially, I thought particle system wasn’t that special. Afterall, in my head, I thought particles were just “particle” like sparks or snow. The only reason I chose it was that I find it quite diverse when it comes to its usage as compared to the other topics.

**While doing the project:**

While researching about what kind usage particle system has, I was amazed by how much I underestimated particle system. It was use in object, simulating living creatures and even adding more life into foliage.

In one of the slides where it was **features a monkey**, I was the one who did it because at that time I really did think that every grass strand and object were game object. It really blew my mind when I thought that some game object could be particles in disguise. It also fascinates me how this particle could also do **collision** with game object so that it acts like game object without giving the hint that it is a particle.

This project has also given me an idea on how many of the games are able to show so many details like foliage, enemies and even miscanlanous objects.

While working on this project, I manage to understand and learn how to use unity VFX graph. Though it was a hard process as this was my first time learning visual scripting. I find it fun and enjoyable once I manage to get the results. I had to do multiple trail and error just to get satisfactory result and even optimisation to make sure that I can render more grass strands.

**After doing this project:**

I felt like I was glad to do this group project. It given me a lot of insight as to how particles can be used in a variety of ways despite its name. I also manage to come out of this project with knowledge on how to use VFX graph which can be helpful for future game project I be making.

This project has also given me a path to start trying out new things. As VFX graph is still limited in the ways it can do due to Unity APIs, it makes me wonder if I can make my **own implementation** of a **particle system** in unity. This feels more plausible as I know that I can code it out using compute shaders and noise map. So, it might be a fun side project for me to try out during my holidays as I try to learn about particle system.

Overall, this project was quite fun and fulfilling. It has equipped me with knowledge about how to use particle systems and how to implement them.

1. **GMAPS reflection**

**What do I think about this subject?**

I love this subject a lot. I always look forward to the worksheets given as it is quite fun to complete. The only complain was not having more difficult question to solve for each worksheet.

The knowledge that GMAPs has given me an idea on how to use this concepts into games (like matrix for rotation. Etc). It also help simplify complex concept that is taught (such as the physic formulas) which helps me understand a lot of its use cases. Finally, many of the worksheet and projects given has allowed me to grow through self-directed learning. Unlike the other subject in year 2.1, GMAPs is structure in a way where I can know what I am learning. But it is also dynamic enough for me to explore advance concepts that I think will be helpful for my game development journey.

**Did it help me?**

The subject has help me directly and indirectly in some sense. Well, it did help me learn about some physic and math topics. It also stirs a lot of curiosity in me to want to discover more about more advance concept. As I want to do many side projects during my holiday, this subject give me more ideas on what other things I can make like game engine or even math libraries.

It also helps me indirectly in a few ways. One was how I absorb and adapt to new technologies. During the implementation part for my group project, I was thrown to learn VFX graph within a few weeks. With many time commitments, I was forced to learn a new way of scripting and behaviour within a matter of weeks. This gave me the opportunity to learn and reflect what I did right and wrong when learning new technologies. For examples, one of my mistakes while learning VFX graph was rushing through the tutorials and skimming through the documentation. Though its safe time, I would miss certain key information which would have been helpful for development.

Conclusion

Overall,