Project Announcement

This project will be assessed based on the following four components:

- **Mid-term written report** (Due: Dec 3, team-based, up to 5 points)
- **Mid-term oral presentation** (Temporarily scheduled on Dec 4, team-based, up to 5 points)
- **In-class presentation** (Dec 31, team-based, up to 15 points)
- **Final written report** (Due: Jan 14, **individual**-based, up to 15 points, hard deadline, no extensions)

The total project score is 40+bonus points.

1 Mid-term Written Report

Submit a report summarizing your project progress using the SIGGRAPH template. One **full** page is sufficient for this report, but do not be more than two pages. You need to include the following information:

- The project topic you chose
- Your project goal and the technical points you intended to include
- A detailed schedule for the whole project
- The technical aspects you have finished, including methods used and any visual results (if available)
- Your plan for the remaining technical tasks
- External tools you are using or plan to use

2 Mid-term Oral Report

You are going to attend the TA session scheduled on Dec 4, 5-6, to show your project progress to the TAs **in person** with demos you might have already finished. It is not necessary to have completed all features by this time, but make sure to have a working demo for presentation. Notice that the oral report is an important criterion for the TAs to select orals for the final presentation.

3 In-class Presentation

Each team is required to create a poster and provide live demonstrations for the final presentation. The three TAs and the instructor will evaluate each poster based on the quality of the demo—considering factors like the aesthetic appeal of your rendering, the gaming experience of your interactive game, or the complexity of your simulation—as well as the overall technical presentation and the effectiveness of the on-site Q&A.

Additionally, all enrolled students will have the chance to vote for their favorite projects, with the most popular projects earning up to 3 bonus points. We will also award 2 bonus points for oral presentations. The TAs will select approximately 10 projects for oral presentations based on the mid-term oral report. In addition to the required poster presentation, these selected projects will have the opportunity to be presented orally in class on December 24.

4 Final Written Report

Each individual must submit a final report detailing their personal contributions using the SIGGRAPH template. The report should be at least three pages long and include the following sections: Introduction, Method, Results, Discussion, Personal Contribution Statement, and References. Each report must be written independently—**no shared writing**, even within teams.

In addition, you must submit the source code for the project (a GitHub link is recommended). Whether working individually or in a team, proper commit management is advised to help TAs evaluate your workload. In the final written report, make sure clearly indicate which part of the code you contributed.

5 Projects

In the following sections, the red parts represent the core requirements for this topic. You must **fully implement** these components; failure to do so will result in your project being considered incomplete. It is acceptable if you choose not to pursue the optional parts, but skipping the entire project is **not allowed**.

There are two possible scenarios:

- **Individual**: You must complete the red parts, which will automatically earn you 5pts. Afterward, you may optionally select any combination of the **black parts** to earn up to an additional 10pts.
- **Team**: Your team must complete the red parts, which will automatically earn each individual 8pts. Then, each individual may optionally select a **different** combination of the **black parts** to earn up to an additional 7pts. This means the red parts are completed collaboratively, but only one team member can receive points for each of the black parts.

Overall, we encourage everyone to complete the project as a team. Since the high difficulty of some technical points, we believe that completing 10pts individually is comparable in difficulty to completing 14pts as a team, so the total workload is similar. However, if you choose to work as a team, you are more likely to achieve higher completion and presentation quality, which can help you earn better scores and possibly bonuses during the presentation portion.

For the writing portion, the TA will evaluate the quality of each individual's report. Deductions of 1-5 pts may be applied for issues such as improper structure, unclear methodology, missing experimental results, or insufficient length.

If you wish to implement any advanced techniques in your project, please contact the TA before the **Oral Report** to discuss feasibility and grading criteria.

5.1 Topic 1: Image Rendering

In this section, you are required to implement a renderer, which can be either hardware-based (GPU) or software-based (CPU). The goal is to produce a static image. Note that a path-tracing renderer is required, as rasterization-based renderers will not be accepted. Also, there is a framework that you can choose to use it or not(`github.com/LazyJazzDev/Sparks`).The functionalities you need to implement are as follows:

- **Base:** Implement a path tracing algorithm that correctly handles **diffuse** and **specular** materials. (basic)
- **Scene creation:** Build a custom scene with aesthetic considerations, using geometry that you create from scratch or find online (ensure the source is credited). (basic, tidiness and attractiveness 1pt)
- **Acceleration structure:** Implement an acceleration structure such as BVH (Bounding Volume Hierarchy). This is not required for hardware-based renderers, as the acceleration structure is built-in in that case. (basic, Surface Area Heuristic or another advanced algorithm 2pts)

- **Material:** Create a (non-trivial) custom material. Options include:
  - Transmissive material (basic)
  - Principled BSDF (2pts)
  - Multi-layer material (2pts)
  - Rendering of fur, hair, skin, etc. (2pts)
- **Texture:** Create your own (non-trivial) texture with proper texture mapping. Options include:
  - Color texture (basic)
  - Normal map, height map, attribute map, or any functional texture mapping (1pt for each, up to 2pts)
  - Implement an adaptive mipmap algorithm (2pts)
- **Importance Sampling:** Use more advanced sampling algorithms for path tracing. (Importance sampling with Russian Roulette, multiple importance sampling 2pts)
- **Volumetric Rendering:** Options include:
  - Subsurface scattering (2pts)
  - Homogeneous volume rendering (1pt)
  - Inhomogeneous volume rendering (1pt)
  - Channel-independent subsurface scattering (1pt)
  - Volumetric emission (1pt)
  - Volumetric alpha shadow (2pts)
- **Special Visual Effects:** Options include:
  - Motion blur, depth of field (basic)
  - Alpha shadow (basic)
  - Cartoon style rendering (2pts)
  - Chromatic dispersion (2pts)
- **Lighting:** Options include:
  - Point light and area light (basic)
  - Environment lighting with HDR, such as skybox (2pts)
- **Anti-aliasing:** Implement an anti-aliasing algorithm (basic)
- **Simulation-based content creation:** Up to 2pts

5.2 Topic 2: Interactive Game

In this section, you are required to design an interactive game. The type and content of the game are flexible, but you must avoid using highly polished, complete game engines like Unity or UE5. A good starting point would be using Three.js. (If the engine you choose results in significantly less code than Three.js, you should be concerned about whether the workload meets the project's expectations). Additionally, the use of any engines that involve visual programming is strictly prohibited.

The following functionalities are required for your game:

- **Graphics Assets**:
  - Scene layout: reasonable object geometry, textures, and materials (basic)
  - Environment lighting (1pt)
  - Synchronized audio (1pt)
- **Animation**:
  - Kinematics:
    - Include animations, whether simple keyframe animations or more complex ones (basic)
    - Articulated objects: such as animals or plants, with rigging and skinning (up to 3pts)
  - Dynamics (considering object interactions):
    - Fluid simulation: e.g., water or wind (up to 2pts)
    - Soft/deformable objects (up to 2pts)
    - Collision handling (up to 2pts)
- **Interactive Control**: Ensure natural, intuitive, and smooth motions with controller inputs:
  - Control of the main game character (basic)
  - Camera motion control in either third-person or first-person view (basic, not required if developing a 2D game where the character stays at the center)
  - Implement suitable camera shakes during collisions or movement to enhance immersion (up to 2pts)
- **UI Design**:
  - Proper game start and end interfaces (basic)
  - Additional auxiliary interfaces (up to 2pts)
  - User-friendly layout with visually appealing design (1pt)
- **Advanced Rendering**:
  - Special visual effects such as motion blur, depth of field, etc. (1pt for each, up to 2pts)
  - Path tracing rendering with PBR materials (up to 3pts)
- **Other Advanced Game Features**:
  - Multi-player mode (up to 3pts)
  - Support for various controllers, such as not only for mouse and keyboard, also for real controller (up to 2pts)
  - Customizable character appearance (up to 2pts)
  - Increasing difficulty levels throughout the game (up to 2pts)
  - Save and load system, allowing players to save and resume their progress (up to 2pts)
  - Online multiplayer system (up to 4pts)
- **Completion**:
  - A fully functional, playable game (basic)
  - User manual or instructions (basic)
  - Easy installation or online access (up to 2pts)

5.3 Topic 3: Collision Simulation

In this section, you are required to simulate the collision of different types of objects and render the simulation outcome with an industrial renderer. And please note, all external packages for the simulation are not allowed (i.e., the simulation part must be implemented by yourself). The functionalities you need to implement are as follows:

- **Basic Types**: Choose at least one (basic). Options include:
  - Rigid Body
  - Deformable Body
  - Cloth
  - Fluid
- **Coupling**: Implement physical interaction between different types of materials. (5pts max)

	Solid	Fluid	Cloth
Solid	up to 3pts	up to 4pts	up to 4pts
Fluid	up to 4pts	up to 3pts	up to 5pts
Cloth	up to 4pts	up to 5pts	up to 3pts

- **Geometry:**
  - Trivial geometry (basic)
  - Complex (mesh based or equation based) geometry (up to 2pts)
- **Acceleration:** Need to analysis the bottleneck and prove your acceleration ratio.
  - Algorithm acceleration (up to 4pts). For example:
    - Acceleration structure (for collision detection)
    - Advanced numerial methods (for solving arithmetic equations)
  - Multi-thread acceleration (up to 3pts)
  - GPU acceleration (up to 5pts)
    - Mutually exclusive with multithread acceleration
    - The score will depend on your workload and acceleration ratio. For example, using Taichi's GPU acceleration (e.g., `ti.init(arch=ti.gpu)`) will result in a really low credit on this part.
    - CUDA and NVIDIA Warp are recommended.
- **Control:**
  - Customized scene configuration (up to 3pts)
  - Interactive scene: Options include:
    - Fixed operation in procudure (up to 2pts)
    - Interactive in real-time (up to 2pts)
- **Rendering:** Use an industrial renderer to produce a nice video :) (basic)
  - You still need to have a taste, even though this is not a rendering work. Illustrating with raw physical elements, such as pure particles and strings, would receive a really low credit on this part.
  - Both offline rendering and realtime rendering are acceptable.
  - Although the rendering part is strictly required, you can earn some additional points based on your rendering results. (up to 4pts)

5.4 Topic 4: Graphics System for Robotics

We are building an all-in-one graphics system, LongMarch, for robotic applications. It requires a variety of techniques, covering rendering (rasterization, ray-tracing), physics simulation (solid, deformable bodies, fluids, etc), geometry processing, systematic engineering, etc. If you want to take a challenge in advanced graphics technologies, this is your opportunity!

The requirement of this project is different from the previous three, we are not asking you to complete a whole project. Instead, you are building components for LongMarch. And your grade will be evaluated with your contribution. The whole project will work with the following steps:

1. We are releasing challenges on the issues page.
2. You choose the issues you are interested in, and work on it.
3. Once you achieved the requirement of your selected issue, submit your code, test results, and technical report with a pull request (PR).
4. Once we verified your PR, we will merge your changes in that PR into the main branch, and you will get points.
5. Then you can move on to the next issue or more.

Grading: Each requirement in the issues is assigned with a number of point. Once you submit a pull request, and we confirm that your work indeed achieved the associated requirement, you will get all the points of that requirement. Furthermore, if your PR is accepted and merged, you will get extra 50% points of the requirement. e.g., if an requirement worth 10 points, and your PR is accepted, you will get 15 points in total. You can submit issues by your own, we will check on it an assign it with a number of points. The issue list may update anytime.

Since this project is a big team work, you are expected to keep a close communication with us. Join the following WeChat group if you are interested in this project.

Feel free to contact Zijian Lyu (zijian.lv@hotmail.com, @LazyJazz on WeChat) if you have any question. We are looking forward to see you in this project.

P.S. The LongMarch framework is still NOT READY, the release time of main framework and basic document is estimated by the end of National Day Holidays. We can discuss about the general design, the infrustrature and theoretical part during the wait for framework.

5.5 Topic 5: ProjectX

If none of the project topics mentioned above align with your interests, you have the option to propose a new project idea. In order to do so, you must create a grading sheet outlining all the technical aspects planned for your project, similar to the other projects listed earlier. Then, you need to make an appointment with TA before October 15 to obtain TA's consent. **You will only be allowed to pursue your own project idea if it is approved by the TAs**. Otherwise, you will still be required to select a topic from the aforementioned list as your final project.

群聊：ACG Project 4 开发交流群

该二维码7天内(10月11日前)有效，重新进入将更新

Figure 1: Wechat group QR code