Introduction to Graphics Programming and its Applications

繪圖程式設計與應用

Final Project

Instructor: Hung-Kuo Chu

Department of Computer Science

National Tsing Hua University

CS5507

Final Project

- Announce date: 2020/12/14
- **Sheet deadline: 2020/01/06, 23:59**
- **Project deadline: 2020/01/10, 23:59**
- **Demo: 2020/01/11 (in class)**
- **You get 0% if not present for demo**
- **45%** of final semester score
- Please submit your project & report via **FTP**

Objective

1. There is no limitation about final topics.
2. Get some models and make them well-arranged.
3. Using cool shaders and create a fantastic scene.
4. Implement rendering algorithms.
5. Using advanced rendering techniques.

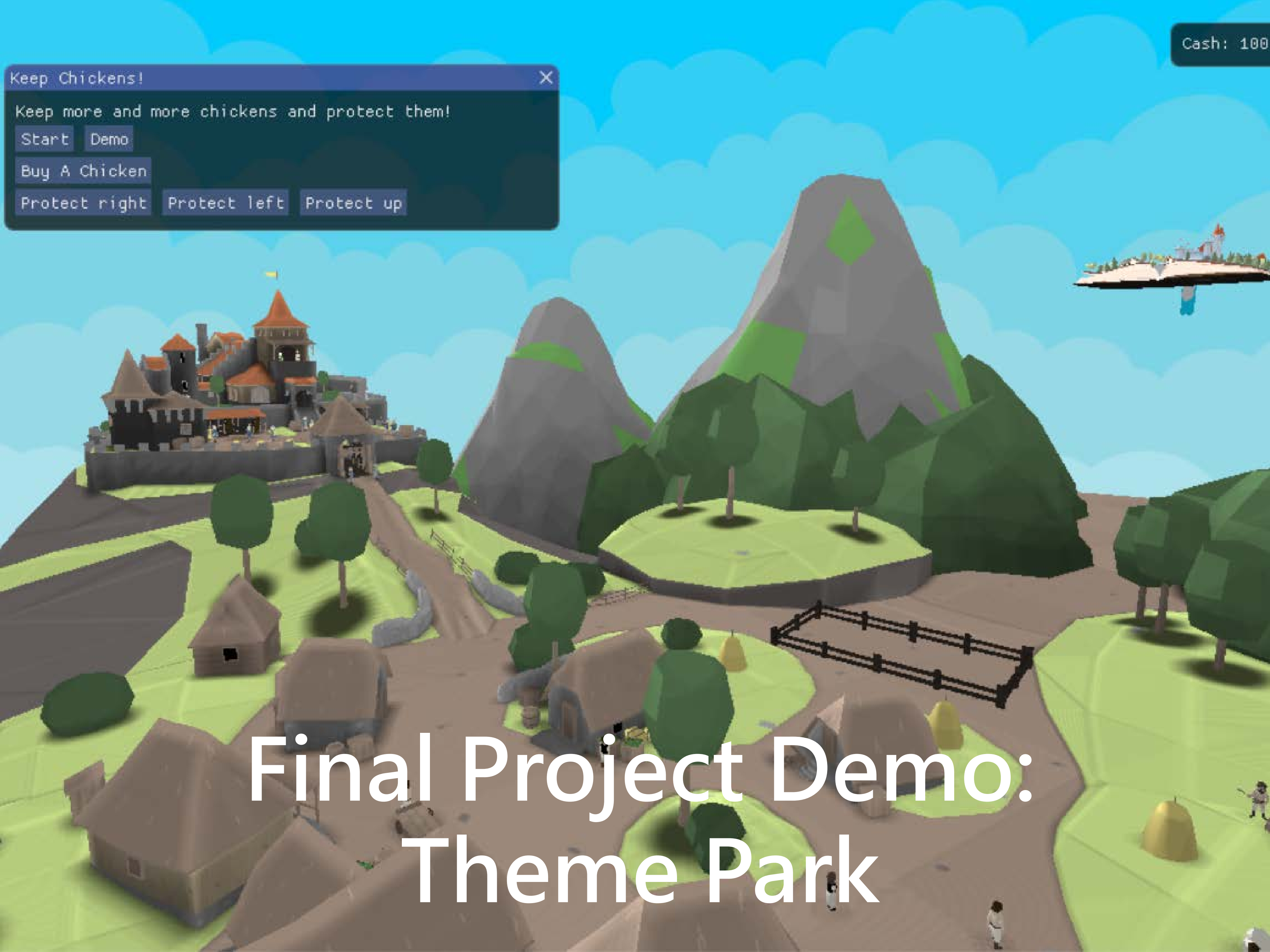
Notice (Content)

- Please make your scene reasonable, combining things arbitrarily WILL NOT get a high score.
- You can deliver your works either as an animation clip, a game, or an interactive program.
- You are strongly encouraged to use advanced skills that are not covered in the lecture.
- Make sure TAs are able to verify those rendering effects claimed in the demo and report, such as comparison bar or switch.

Notice (Demo & Team Up)

- Present main functions and features ONLY in demo due to the limited time.
- Make sure the program is executable in TAs’ machine.
- Team up and work together (up to 4 members in a team), TAs will evaluate personal grade from peer assessment (互評)

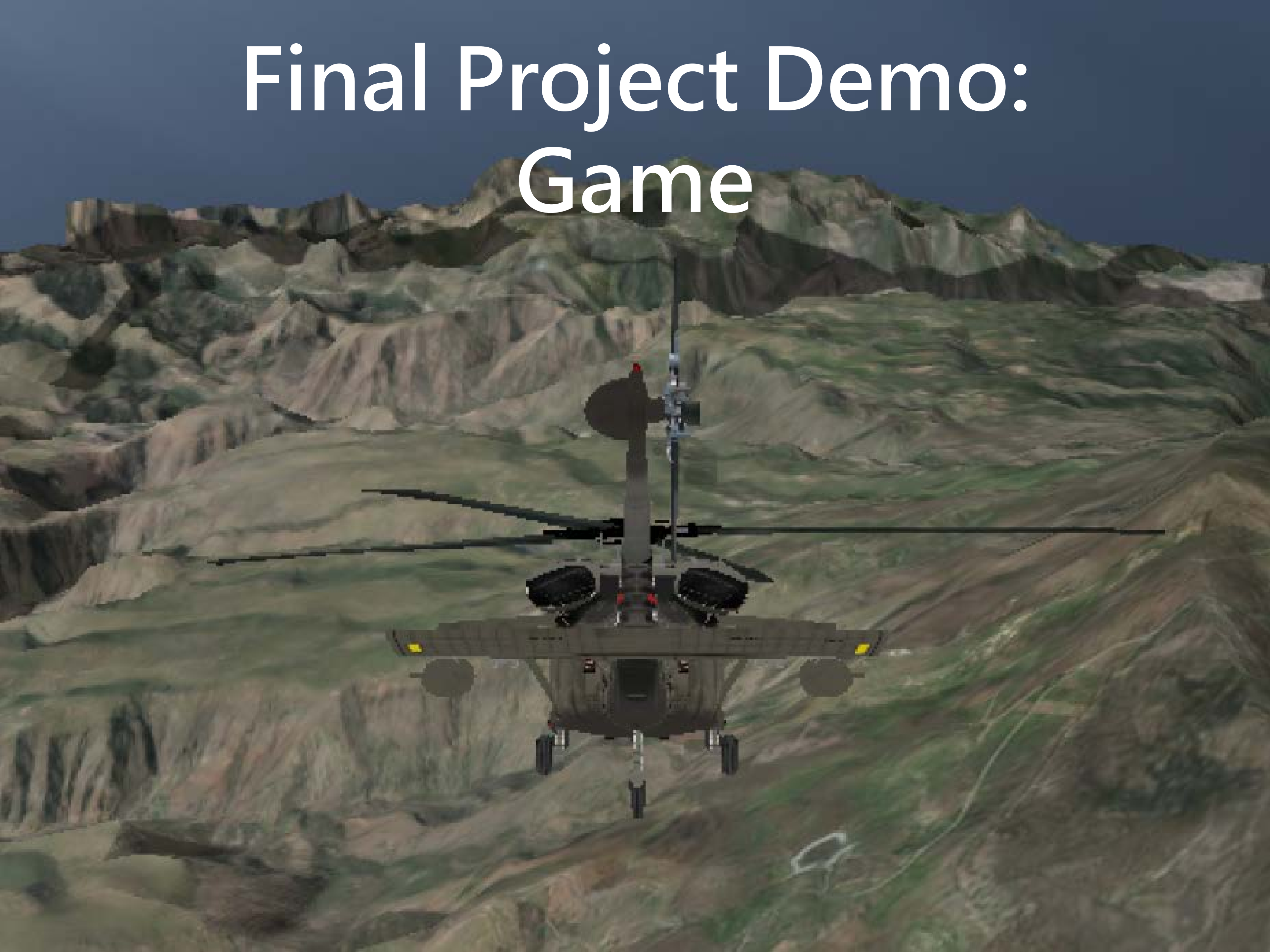
Final Project Demo: Theme Park



Keep more and more chickens and protect them!

Final Project Demo: Theme Park

Final Project Demo: Game



BASIC (80%)

Evaluation

Basic Item	Score
Subjective	20%
Lecture Topic	60%
Total	80%

Subjective

- Consistency
  - Using 3D models with the same or similar style.
  - Place right object in the right place and do the right thing.
- Diversity
  - Try combining some basic tricks to your scene. (Lighting, Skybox, Framebuffer processing…)
  - Different types of 3D models (human figure, animals, man-made objects) are welcomed.

Subjective

- Interactivity
  - Users are able to control some function via keyboard / mouse / UI / built-in menu / trackball / scene routing
  - Add some sound effects (optional)
  - Easy physical interaction (additional bonus for advance item)
- Robustness
  - Won’t easily crash
  - No weird behaviors

Subjective

- Correctness
  - The whole scene is rendered correctly (Geometry and Texture).
  - Bug-free.
- Aesthetics
  - Using high polygon meshes.
  - Using low polygon meshes with normal mapping (preferred).

Subjective

- Presentation
  - Effectively demo the rendering effects under the limited time
- Report
  - Each rendering effect has a pair of ON and OFF
  - Some good references found

Lecture Topic

- Basic **[5% per tech.]**
  - Each technique should be bug-free and correct; otherwise, it will get 0 point
- Rendering technique **[10% per tech.]**
  - The score of each technique in this section will be depend on the completeness

Basic

Basic Item	Description
Blinn-Phong Shading	Calculate in the same coordinate system and vector direction to get correct shading effect.
Framebuffer Postprocessing	Apply the image processing technique to the framebuffer, like bloom.
Skybox	Render the skybox in perspective view with correct cubemap order.
Shadow Mapping	Implement some detail of shadow mapping and render it correctly.
Environment Mapping	Make some object metallic and reflect the environment correctly.

Basic

Basic Item	Description
Normal Mapping	Apply correct transformation to get the detailed shading.
Toon Shading	Not just quantize the shading color, but apply it with ramp texture.
Deferred Rendering	Show many lights environment or each feature in the G-buffer.

Rendering Technique

Technique	Description
Screen-Space Ambient Occlusion (SSAO)	Implement and change the parameters such that AO is reasonable.
Grass	Not only instance a lot of static objects, but also with animation.
Terrain (Height Map)	Perform detail in your terrain like trees, rocks, generating your terrain with height map rather than an existing mesh.
FBX Animation	Animate the character with meaningful clip, like walking, running in the scene.

ADVANCED (30%)

Advanced

- Lecture Topic **[10% per tech.]**
  - The score of each technique in this section will be depend on the completeness
- Advanced Topic **[20% per tech.]**
  - The score of each technique in this section will be depend on the completeness

Lecture Topic

Technique	Description
Water	Water with wave (vertex displacement), refraction, reflection.
Particle System	Implement some fantastic particle system (rain, fire, smoke , spark, fluid)
Cascaded Shadow Mapping	Split the viewing frustum into several levels to produce the depth map.
Ray-Marching-Based Screen-Space Reflection	Apply reasonable material with reflection effect.

Advanced Topic

Advanced Item	Description
HDR Tone Mapping	Implementing a tone mapping algorithm that is useful to your final project.
Image Based Lighting	Try to implement any image based lighting algorithm.
Environment Map Lightness Adjustment	Process environment map by spherical harmonic function to adjust a global illumination.
Physics Simulation	Physics system works correctly, collision detection, character walk on terrain.

Advanced Topic

Advanced Item	Description
Game Mechanism	Make it a game, a full operation (start->menu->end->restart), and some game play elements (defeat monsters, collect treasure…).
Depth of Field	Adjust the blurring according to depth buffer and specified focal length.

Advanced Topic

Advanced Item	Reference Link
Stochastic Screen-Space Reflection (SSSR)	Tomasz Stachowiak at Electronic Arts in SIGGRAPH 2015 course
Real-Time Volume Rendering	A new alias-free sampling algorithm for the camera-in-volume case
Image-Space Horizon-Based Ambient Occlusion (HBAO)	Advanced Ambient Occlusion Methods for Modern Games
Reflective Shadow Mapping	Original paper

Advanced Topic

Advanced Item	Reference Link
SSAO Tricks and Denoiser	Strided blur and other tips for SSAO
SSR Denoiser	Screen-Space Reflections Shader v0.7
Spatiotemporal Antialiasing	Blog post
Shadow Map Antialiasing	GPU Gems Chapter 11

More Ideas

- Advances in Real-Time Rendering in 3D Graphics and Games in SIGGRAPH Course
- GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics
- Shadertoy: Build Shaders, Share Them, and Learn from the Best Community

Report Spec.

- File name: **Team#_Report.pdf**
- List all the team members
- Be specific about the (advanced) techniques you have done in your project.
- Present each (advanced) technique by showing a screenshot.

Peer Assessment (互評)

- TAs will give each team an assessment table as shown below on the demo day.
- All the team members have to sign the form to acknowledge the claimed contribution.

組員	學號	工作	貢獻度
朱彥睿	108062573	蒐集模型、安排場景....	25%
林立雯	105042004	UI、物理系統...	25%
梁景勛	109062544	系統整合、Game、Demo...	25%
潘仁逸	109062571	Report、Idea...	25%

Peer Assessment (互評)

假設組員人數為 N

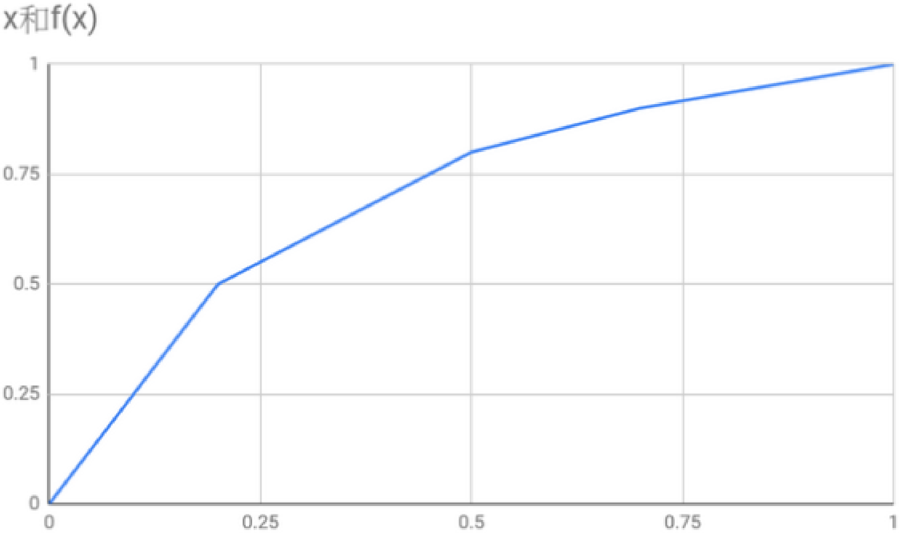
每位組員的期望貢獻度為 Rd = 100 / N

而實際貢獻度為 Rt

定義比例 x = min(Rt/Rd, 1)

再由 x 用下圖的函式計算分數該要乘上多少 (f(x))

使得最後組員成績為 (原始小隊分數) x (f(x))



Team Assessment (小組自評)

- Each team should submit the following link (google sheet) so that TAs can know the implemented items before demo
  - https://forms.gle/gtosW6CH5inh8VGh9

MD5 checksum

- 除了作業本身外，請透過 MD5 獲得作業 checksum 後填入 google 表單
- 如遇各種原因無法在作業期限前完成上傳的同學，我們將比對 checksum。若 checksum 一致則不算遲交。
- 若有多個 checksum 則取時間最晚的為主。
- MD5 online generator
- MD5 checksum 登記表單
- MD5 checksum 登記查看
- MD5 使用方法

Libraries

- Assimp
  - https://github.com/assimp/assimp
  - Load all kinds of model, .OBJ, .FBX, …
- Nanovg
  - https://github.com/memononen/nanovg
  - OpenGL friendly GUI library
- ImGUI
  - https://github.com/ocornut/imgui
  - Very powerful GUI library, works with OpenGL

Libraries

- DevIL
  - http://openil.sourceforge.net/
  - OpenGL-inspired Image loading library
- STB
  - https://github.com/nothings/stb
  - Useful single header libraries
  - std_image.h: image loader
  - (Note: nanovg uses std_image.h too)