# SDL2: BULLET HELL

GAME PROGRAMMING WITH C++
KRISTER@FUTUREGAMES.NU

# WHAT

This is a group assignment with an individual part as well.

The **group assignment** is to create a 2D shoot 'em up game using C++ and the SDL library.

https://en.wikipedia.org/wiki/Shoot_%27em_up

https://youtu.be/R0mb62phlKU

The **individual** part is reflecting and explaining the following:

- For each system you created or helped develop.
    - How does the system work and interacts with the rest of the game?
    - Strengths and weaknesses of your code/system.
    - How could you improve the system?
- Teamwork
    - What were some of the challenges during this project?
    - What mistakes and failures did you or your team have?
    - What were some of the highpoints of the project?

# HOW

For this project you must make the game in **C++** and **SDL2**. You may add any other accompanying library as you see fit.

During the development keep the reflection part in mind, take notes to help you summarize it in the end for the individual part. Also keep in mind if this is something you want to use for your portfolio.

# SDL2: BULLET HELL

GAME PROGRAMMING WITH C++
KRISTER@FUTUREGAMES.NU

# SUBMISSION

Use a public **GitHub** for the repository and send the link via the **Canvas** assignment. You can write the individual part directly in the **Canvas** assignment.

Deadline: 2019-12-01; 23:59

# EXAMINATION

This assignment is individually graded for **Icke Godkänt (IG)**, **Godkänt (G)** or **Väl godkänt (VG)**.

The below requirements are for one programmer to implement for the corresponding grade.

**Godkänt (G)**

- Consistent code
- Create at least one system using C++.
    - Examples: Highscore (savable/loadable), simple GUI, simple collision system, sound manager, tile map.

**Väl Godkänt (VG)**

- High quality code
    - Optimization, error handling, clean
- Inheritance
    - Use inheritance, this could be an entity system, equippable weapons or replaceable
    - Classes and methods should have clear responsibility: Single-responsibility Principle.
- One advanced system (talk to teacher with your suggestion).
    - Examples: Pathfinding, behavior tree, serialization, collision system (pixel perfect, rotations), GUI.

# ASSIGNMENT EVALUATION

Any feedback is greatly appreciated.

Do you feel you had the necessary knowledge to tackle this assignment?
Was this assignment valuable?
Was this assignment too easy or difficult?