# COMP30540 "Game Development"
# 2nd Game Programming Assignment (of two in total)

Announced Thursday 31 January 2019. **Due Monday 15 April before 11am.**
This assignment is given a letter grade and is worth 25% of the marks for the module.

Note that plagiarism is an offence taken seriously in UCD, Science, and CS. Do not present work as your own that is provided to you by someone else or copied from them. Acknowledge the sources of materials you use fairly, such as photographs or stock images or 3D models. Do not allow or encourage others to copy your own work.

## Drone Destroyer

Use any free Game Engine (*except* any version of GameMaker) to develop a 3-dimensional game for laptop (PC or Mac or whatever) according to the description below. You will be asked to demonstrate (to lecturer and assistant) the play of your game, and to submit a short report and any code (scripts) you have written to make it work. The demonstration should be expected to take around 15-20 minutes including talking time. You should also provide a low-resolution (for low storage space) 2-3 minute screen recording of the game being played by you.

It is not necessary for the game to run independently, although it would be nice if it did. It is quite acceptable for the Game Engine development tool to be required to run the game.

**The storyline of the game**
Criminals have been disrupting airports by flying drones that could cause great damage if they were hit by an airliner taking off or landing. Your airport has invested in a high spec drone designed to cause lesser drones to crash, and employed you to pilot it remotely. Your weapons are designed to tangle up in the rotors of other drones, making them unstable and unflyable. Unfortunately the criminals' drones are now equipped with similar weapons, and their operators will try to take your drone down too. You score points by making their drones crash. Your own drone is quite resilient because it has 12 rotors and so needs to be hit several times before it will crash. You also can rely on airport personnel to remove any entanglements so you can fly again, but you have to land safely near enough to them for them to do that. You don't need to worry about running out of power, you have big batteries that can be instantly replaced by those airport workers. You also have to avoid crashing into any airliners yourself, and avoid hurting airport staff who are there to get you going again. If you are made to crash, it is game over – or lose a life, if you prefer. If you kill enough drones, you get a better drone for the next time, but you have to face more criminal drones at a time.

**The display**
The player's view should be from a camera mounted underneath the rotors of your drone. You can see some of those rotors, and the part of the world in front of the camera: there may be an airport runway, an airliner waiting to take off and/or another one coming in to land, airport workers beside the runway (a dangerous job to be sure), and criminal drones. You could put a fence around an area to limit the part of the world that is interesting. You see the missiles that you have shot and those shot at you by criminal drones. Ideally you should also have a minimap to allow you to sense in less detail what is all around you. The number of hits your drone has taken, the number (or accumulated points value if you prefer) of criminal drones destroyed should also be

shown. As you fly, your view of the world may turn in various ways, but you will always be generally upright unless you are crashing.

**The controls (W, A, S, D, arrow buttons, Return, Space)**
The specific controls mentioned are just a suggestion, you may use different keys if you want to.

You can speed up forwards, to a top speed, and slow down or go backwards, to a top speed. You can turn to left and right, and to a limited extent you can tilt up or down. You do these things with W (forward acceleration), S (slow down), A (turn a bit to your left), D (turn a bit to your right). W and S make you continue to speed up or slow down for as long as they are pressed, up to a maximum speed or down to a stop and no further. A and D make you turn left or right at constant speed for as long as they are pressed. Opposite keys at the same cancel each other out, but it should be possible for example to accelerate while turning. Stopping to hover is expected to be a tricky manouevre.

Arrow keys make you tilt your whole drone including missile launcher up and down, or just the missile launcher left and right, relative to the direction you are looking. A tilt of more than 45 degrees is not possible, nor is a left/right deviation of more than 30 degrees.

The space bar makes you launch a missile. Because of gravity you will have to aim the launcher above the point in 3D that you want to hit. The missile consists of two parts: a blob of ice, that will melt and pose no danger to airliners on the runway; and a trailing thread. The idea is to hit a drone's rotors so that the thread tangles one of them up and causes it to stop.

Animation of a thread flying through the air is an important part of the assignment.

**Gravity.**
When you launch a missile, its path should curve downward. If you then turn or tilt, the missile's path should not turn or tilt with you. Its speed and direction should be changed according to the pull of gravity.

**AI criminal opponent**
The opponent should be able to do most of the things you can do, moving around and trying to cause your drone to crash. Its main mission however is to threaten the airliners, simply by being in their way, so it won't chase you too far from the runway. The criminal's drones are not as good as yours, they have three or four rotors and so only need to have one rotor disabled to make them crash. Airport workers won't mend them. The missiles shot at your drone should be very visible to you. It should try to aim missiles at the point where it thinks you will be when the missile arrives, rather than where you are at the moment it fires. It will probably be able to perform accurate calculations and thus carry out precise manoeuvres much better than you as player, so you as programmer should try deliberately introducing some random inaccuracy into its aim and/or timing of actions - they are not perfect shots any more than you are
.

**Levels of difficulty**
You should provide for a menu system allowing the player starting a game to choose between at least two levels of difficulty (and offer Help and Quit options). A more difficult level might feature *for example* a more nimble and/or more accurate opponent, more criminal drones present

at once, better criminal drones requiring two hits each before crashing, fewer airport workers to mend your drone

**Suggested embellishments**
Allow 'Q' for Quit. Add sound effects, e.g. a whoosh when you launch a missile, sounds of airlines taxiing or landing, rotors getting tangled up. Make the view rotate when you fall. Use some simple 3D model you built yourself, e.g. a spare battery with each worker. Make your direction of gaze wobble a bit. Make the probability of falling when travelling backwards depend on speed and time elapsed.

*You are most definitely **not** expected to achieve all of these embellishments.*

**Experiment**
Experiment to find satisfying values for various quantities:

- speeds (of angular accelerations; of forward/ackward acceleration; of missile)
- constants (limits on speeds of travel and of rotations; size of runway; size and number of criminal drones; number of hits your drone can survive; how hard you can land beside a worker)
- formulas (for changing direction and speed; for inaccuracy of AI's aim)

**Artwork**
Do not waste a lot of your time on fancy artwork or animation. The point of the assignment is to develop and demonstrate an ability to develop a 3D game, not to produce a highly polished game. Animation of flying thread is highly desirable, a little 3D model building is desirable, but not a lot.

**Report**
Write a report on what you have achieved, include screenshots of the menu system and a few (static) highlights from play of your game. Describe any difficulties you had, whether you solved them or not. Describe how you think the game might be further improved if you had more time to spend on it.

Apart from screenshots, your report should not exceed four pages of text unless you have really interesting things to say about experiments that led to your choices for quantities and formulas to make the game satisfying. *You do **not** need to write as much as nearly four pages.*

# Submission Do's

***Do** put your name and student number in the first few lines of the report, and in comments in the first few lines of each source-code script file you write. If you get your chosen Game Engine to create scripts for you, extract them - using screenshots to do so if there is no better alternative - and submit them too.*

***Do** use moodle to submit a .zip containing*

- *your source code files (including Engine-generated ones)*
- *your report*
- *a 2-3 minute low-resolution screen recording of you playing the game*

***Do** make that zip file unzip into a folder bearing your name. Temporarily rename your own copy while making the zip file to achieve this.*

***Do** print and sign an assessment submission form: scan it or photograph it, and include it in your zip file.*

# and Don'ts

***Do not include*** anything that is not either source code, your report, your recording or your submission form. In particular, do not submit an executable for your game, or a compiled form of your game code.

***Do not** present your writeup as anything other than pdf. Avoid, for example, .doc and .docx and .odt and .rtf.*

***Do not** submit a .rar, .jar, .7z, or any other form of archive except .zip.*

***Do not** place folders inside folders, even within a .zip*, just one folder of files.

***Do not** provide multiple versions of your code.*

***Do not** send an email to submit your assignment, use only the moodle.*

***Do not** present lines of program code or comment having more than 100 characters.*

Each of the above *don'ts* violated may attract a penalty of 5%.

***Do not** give your submission an awkward filename. Just use your name for it.*

Standard UCD penalties for lateness will apply: if submitted up to one week late, 10% will be lost (turning 75% into 65% for example); if submitted even later but within two weeks, 20% will be lost; if not submitted within two weeks, barring extenuating circumstances, all marks will be lost.

It is far better to submit something, even if incomplete and/or late, than to submit nothing at all.