

COMP30540 "Game Development"

1st Game Programming Assignment (of two in total)

Announced Wednesday 23 January 2019. **Due Monday 4 March before 11am.**

This assignment 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. Do not allow or encourage others to copy your own work.

Egg Collectors

Use GameMaker (free trial version) to develop a 2-dimensional game for laptop (PC or Mac or ...) according to the description below. You will be asked to demonstrate (to lecturer) 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. The screen recording software that is part of GameMaker may not be the best choice.

It is desirable to make an executable so the game can run independently.

The storyline of the game

You are controlling two workers who are trying to catch eggs as they are laid by a collection of chickens arranged in a line. As a chicken is about to lay an egg, it visibly changes, and you have to get a catcher to the right area to collect the egg in his/her basket. If a basket gets too full, new eggs are likely to break existing eggs, so new hay must be got on top of the old eggs. If the two catchers run into each other, they are likely to drop the eggs from their baskets. As controller, you must move the catchers to places where eggs will drop and where hay will be added. Get as many eggs within a time limit as necessary, and be promoted to a better job where the eggs are laid more frequently and/or more quickly; or be fired from your job (game over) if too many eggs get broken.

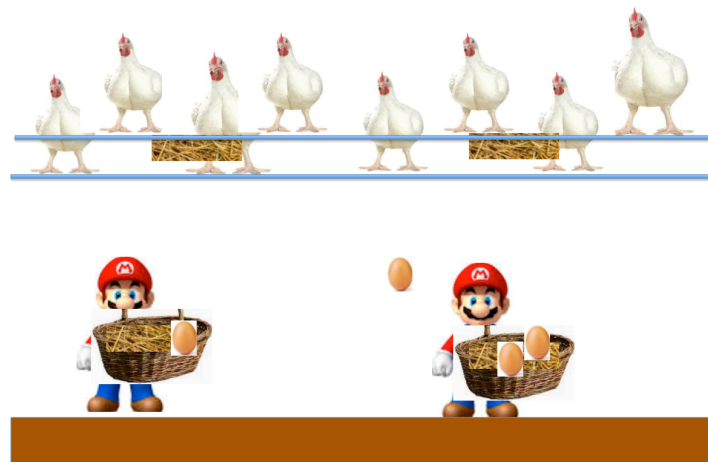


Figure 1: A really crude sketch of the important elements to be displayed with ground, workers holding baskets that may contain hay and may contain eggs, an egg falling, rails on which chickens stand, places where hay drops on command, chickens – of which two are bigger signalling that an egg is soon to be laid.

The display

The view should be from the front. The catchers should be shown and should be movable. The chickens and the hay machines cannot be made to move. Chickens take a random amount of time between laying an egg and feeling another one coming on. Figure 1 above shows roughly the desired layout.

The controls

You can move the workers to the left and right with keyboard keys such as Z X C V M , . / (on my keyboard those are on the left and right of the bottom row, excluding space bar etc.) The group Z X C V make the left worker move, fast or slow and left or right; likewise the group with M , . / for the right worker. Pressing the space bar causes new hay to be dropped.

The action

When a chicken is going to lay an egg, it should change somehow: pulse between a larger and a smaller size, or change colour, or something like that. When it drops its egg it goes back to normal.

When you (the player, the controller) press one of the relevant keys, one of the catchers moves – slow or fast – in the direction of the pressed key. Z = left catcher move fast leftward (from your point of view), C = left catcher move slowly rightward, and so on. The catchers cannot be made to stop except by colliding with each other or reaching the screen edge.

If an egg lands in a basket, it might break other eggs already there. The more there are, the more likely a breakage. Getting a new layer of hay protects the eggs below. You don't need to show eggs either on or under hay, and you don't need to calculate collisions between new eggs being laid and unprotected eggs on the hay in the basket. Breakage is just a matter of probability.

When you press the space bar, both locations above drop a new layer of hay, if a catcher is not properly positioned they will miss it and it will simply vanish.

If the catchers collide with each other, they lose half their unprotected eggs, or all of them if both are moving at top speed.

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 smaller basket, more sluggish response to keys for moving the catchers, eggs being laid more frequently, more chickens, greater chances of eggs getting smashed.

Suggested embellishments

Find appropriate images. Have a scoring system and a panel displaying the score. Add sound effects – clucks, egg splats for example. Add some background imagery that does not interfere with the player's ability to see the chickens and hay and catchers. Create a simple animation of the catchers' feet moving. Have the time between eggs reduce gradually as progress is made through a level. Make catchers change speed and direction gradually while a key is held down, rather than going from one velocity to another in an instant. Make eggs and hay accelerate downward.

You are encouraged to add further features to the game that you think will improve it, perhaps based on your experience of game playing. Beware however of spending too much time on improvements.

Experiment

Experiment to find satisfying values for various quantities, such as for example:

- speeds (of egg and hay falling, of catcher)
- formulas (for picking which chicken will lay at what time, for breakage of unprotected eggs)
- constants (how much time must a chicken rest before it could possibly lay its next egg, how many eggs can get broken before you lose)

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 use GameMaker, not to produce a highly polished game.

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 the events (such as key presses and key releases, and timers) that make your game work. Apart from some screenshots, your report should not exceed three 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.

Advice

The Moodle submission area allows you to upload up to 20Mb. This should be sufficient for the executable, the report, and a screen recording. Recordings of ~200Mb are not appropriate and I don't want to be sent links to Google Drive where I can view them at my inconvenience!

Standard UCD penalties for lateness will apply: if submitted up to one week late, 10% will be lost (turning 62% into 52% 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.

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. Write scripts in GameMaker's GML rather than using its drag-and-drop facility, and submit them too.

Do use moodle to submit a .zip containing

- *your source code files (including GameMaker GML scripts)*
- *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.

Do not present your report 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.

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

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