Reactive Procedural Level Generation

COMP3180 RESEARCH REPORT

Name: Edgar Murga Garcia De Leon

Student ID: 47094133

Topic Category: Procedural Generation

# Introduction

Total word count (excluding final reference list) must be between 1000-2000 words to avoid penalty.

Procedural generation is a technique used in game development to create content algorithmically rather than manually. This is achieved by mixing rules, algorithms and randomization to generate content during runtime. This topic is of great relevance for advance game development as in today’s industry procedural generation is used in a variety of fields to reduce the amount of manual labour needed for some of these fields. These include (not limited to):

* Level/Terrain Generation
* Character Dialogue & Animations
* Object Instantiation & Loot Systems

While there are a lot of fields in which procedural generation is beneficial for game development, I will only be focusing on level generation (and population but more on that later) with an emphasis on level generation beneficial to a roguelike video game.

For context, a roguelike is a subgenre of video games which comes from the video game *Rogue*, and ever since 2008 through the Berlin Interpretation **Random Environment Generation** is one of the most important factors of roguelikes [1]. Therefore, I plan to link my learnings with the need for procedural generation in roguelikes to create a procedural level generation algorithm which would make a roguelike game feel reactive to real time player conditions.

# Literature Review

A Literature Review on your particular advanced game development topic covering 3 to 6 quality sources. This can include both academic and industry sources as well as community sources of sufficient quality. Technical prototypes that you have already developed can also be described (with supporting images) to count as one of your sources. Each distinct source should be clearly identified in the body of your report using [APA citations](https://libguides.mq.edu.au/referencing/APA7thEdition), accompanied by analysis (~200 words for each source) that includes:

* a brief description of the source (1-2 sentences max).
* a (~150 words) summary of what you have learnt from the source (important: do not abstractly summarise the source, rather identify the specific aspects you found the most interesting/enlightening/relevant to better understanding your project area).
* a one sentence statement assessing the usefulness of the source for your final project.

Plan:

* Hybrid Approach to Proc Gen
  + Introduced a bunch of techniques
  + Introduction on Context Free Grammars
  + Intro to cellular Automaton
* Cellular Automaton
* Wave Function Collapse
* Prototype 1

# Project Proposal

An (~500 words) outline of the Project Proposal, including:

## Learning Goals

Specific learning goals identifying the industry-relevant skills you intend to gain through your work on the project. Make sure these are specific and measurable, as you will be expected to evaluate your success in reaching these goals in your final project. You should also identify how your goals are industry relevant and why you’re interested to build these skills (i.e. what personal career goals they relate to). If you need help developing learning goals you can start by reviewing the structure of the [learning outcomes in the unit guide](https://unitguides.mq.edu.au/unit_offerings/156325/unit_guide?full_code=COMP3180_SHFYR_2023#learning-outcomes-section), or review [Bloom’s taxonomy](https://www.coloradocollege.edu/other/assessment/how-to-assess-learning/learning-outcomes/blooms-revised-taxonomy.html) and articles on [using it to develop learning outcomes](https://tips.uark.edu/using-blooms-taxonomy/).

## Deliverables

The specific proposed deliverables (e.g. prototypes, documentation, experiments, analyses) you intend to create for your final project that build on the ideas or address the gaps from your reading. Be sure that each deliverable includes some detail on how its success will be measured (e.g. number of words for a report, supported features for software, number of testers for prototype, quality goal for evaluation, etc.). Evaluation of outputs (e.g. their utility, usability or UX, as appropriate) is a required component of the final project, as is creating a portfolio piece with industry relevance (including a short demonstration video showcasing relevant prototypes), so be sure to identify the individual deliverables that address these requirements.

Your deliverables should include:

* The final project report, including the write up of the evaluation of (at least) your primary deliverable for its intended audience.
* The main deliverable/s, such as software prototype/s, development tool/s, serious game prototype, games user research report, etc.
* The portfolio piece, which should be an industry-ready accessible demonstration of your main deliverable (e.g. a video showing off the game, your prototype, software tool, etc.). If your main deliverable is a report then this may not be necessary, but you should consider how consumable your portfolio piece will be to a recruiter during evaluation of you CV and if you need to provide it in a more accessible format.
* Any additional deliverables, such as documentation for your prototype/tools (e.g. game design documents, technical documents)
* Artefacts from the evaluation of your main deliverable for its target audience, including the instruments you used for evaluation (e.g. surveys, interviews), the raw data gathered, and any work analysing it and preparing it for reports

For each deliverable, you should provide the expected number of hours you will spend on it. You are expected to dedicate **57 hours** of work to this assessment task (including 45 hours of work outside of class time, and an additional 12 hours of project time in the lab during pracs in the second half of semester), so make sure your numbers add up to this total.

Some of the deliverables may be combined into single documents or repositories in your final project submission, but should be outlined here as individual outputs with time allocated to each. Some example project deliverable outlines are included below, with estimated completion hours. These examples are provided as a guideline only, and you should adjust the details (including deliverable descriptions and completion hours) according to the requirements of your project and any preparation and design work needed.

**Example 1:** If you were doing a VR hand-tracking project, your deliverables might be:

* Final Project Report including evaluation write-up (7 hours)
* Unity project containing functional VR hand-tracking prototype in GitHub (28 hours)
* Three-minute demonstration video of VR prototype for portfolio (4 hours)
* Evaluation artefacts in GitHub repo (18 hours)
  + Playtesting instruments and protocol design (4 hours)
  + Recruit/collect usability and UX data for eight users in spreadsheet (6 hours)
  + Document containing collation and analysis of evaluation data (8 hours)

**Example 2:** If you were building a technical prototype for a procedural content generation system:

* Final Project Report including evaluation write-up (7 hours)
* Unity project containing PCG prototype in GitHub (24 hours)
* Three-minute demonstration video of PCG prototype (4 hours)
* Technical user documentation for designers/developers (10 hours)
* Evaluation artefacts in GitHub repo (12 hours)
  + Collection of sample outputs (2 hours)
  + Study evaluating outputs with four level designers (4 hours)
  + Document with collation and analysis of evaluation data (6 hours)

**Example 3:** If you were doing a games user research project on an existing game:

* Final Project Report including evaluation write-up (7 hours)
* Games User Research report (32 hours)
  + Study methodology selection/design, including survey instruments (7 hours)
  + Study execution with 20 participants (10 hours)
  + Observation, analytics and survey data collation (5 hours)
  + Analysis of collected data (10 hours)
* One-page infographic poster highlighting key study achievements/findings (4 hours)
* Evaluation artefacts in GitHub repo (14 hours)
  + Questionnaire evaluating GUR report with four designers/GURs (6 hours)
  + Document with collation and analysis of evaluation data (8 hours)

**Example 4:** If you were developing advanced modelling and/or animation techniques:

* Final Project Report including evaluation write-up (7 hours)
* Blender/Unity project with completed models/animations in GitHub (22 hours)
* Four-minute video of render showcasing produced models/animations (8 hours)
* Technical document outlining explored modelling/animation processes (8 hours)
* Evaluation artefacts in GitHub repo (12 hours)
  + Protocol and instruments for reviewing output/technical doc (3 hours)
  + Evaluation data from six animators/designers in spreadsheet (4 hours)
  + Document with collation and analysis of evaluation data (5 hours)

**Example 5:** If you were exploring methods to perfect game feel in platformers:

* Final Project Report including evaluation write-up (7 hours)
* Three platformer prototypes showcasing unique game feel elements (18 hours)
* Iterative playtesting of game feel/UX in platformer prototypes (12 hours)
  + Playtesting and instrument (survey) design (3 hours)
  + Recruitment/data collection for playtests with five users/iteration (6 hours)
  + Collation/analysis of playtesting data for feedback into design (3 hours)
* Two-minute demonstration video of final prototypes for portfolio (3 hours)
* Document outlining design heuristics for game feel elements (8 hours)
* Evaluation artefacts in GitHub repo (10 hours)
  + Protocol and instruments for reviewing design heuristics/video (2 hours)
  + Evaluation data from four game designers in spreadsheet (3 hours)
  + Document with collation and analysis of evaluation data (4 hours)

# References

A references section that identifies your sources using [APA format](https://libguides.mq.edu.au/referencing/APA7thEdition).

[1] <https://www.makeuseof.com/what-are-roguelike-and-roguelite-video-games/>