

# Common Maze Format v0.1

Mark Swoope

October 31, 2017

## Purpose

This document describes the Common Maze Format (CMF), it is a serialized format for an NxN maze of cells where each cell can contain up to 4 walls.

CMF will be useful because applications that use their own internal data structure for mazes can convert them into CMF and store them into files or strings for other, possibly different applications to load.

## Maze components

An NxN maze can be formally represented as:

$$M = \langle N, S, G, C \rangle$$

The components are:

- N – The total number of cells (must be a perfect square)
- S – The zero-based index of the start cell
- G – The zero-based index of the goal cell
- C – The tuple of cells

Each cell  $C_i$  in  $C$  is a bitmask describing which walls are present in this cell. Each cell can have a combination of the following values:

- 1 – Top wall present
- 2 – Right wall present
- 4 – Bottom wall present
- 8 – Left wall present

## Serialized representation

The serialized representation of an NxN maze is a space-delimited list of it's components where each component is represented as an unsigned, base-10 integer in text-format:

```
<M> ::= <N> <SPACE> <S> <SPACE> <G> <SPACE> <C>
<SPACE> ::= ' ' { ' ' }
<N> ::= <UNSIGNED>
<S> ::= <UNSIGNED>
<G> ::= <UNSIGNED>
<C> ::= <UNSIGNED> { <SPACE> <UNSIGNED> }
<UNSIGNED> ::= <DIGIT> { <DIGIT> }
<DIGIT> ::= '0' ... '9'
```

Here is an example 3x3 maze and it's corresponding serialized representation:

```
+---+---+---+
| S |       G |
+   +   +---+
|   |   |   |
+   +   +   +
|       |
+---+---+---+
```

```
9 0 2 11 9 7 10 10 11 12 4 6
```