Global error-tolerant rules:

* White space “ ” is always tolerated inside map file or within user input.

Note: “\t” is replaced with “ ”.

* Only headline and section PLACES are required in the map file, other sections are optional. Which is to say, as long as there are places in the map, the game is allowed to begin.

Note: 6 component sections totally, include PLACES, PATHS, LIGHTING, ARTIFACTS, KEYS, and LIGHTS.

* Order of sections are irrelevant in the map file. But of course, all sections depend on the PLACES section, so PLACES better comes first when designing a map file.

Other special designs:

* Keys and Lights are all stored in Artifacts, with type 1 and 2 respectively, and normal Artifacts are of type 0. While checking duplicate ID, Keys and Lights are only checked within their own types. //initialized by Dan.

# Error-Tolerant Mechanism of map parsing (Environment.java)

Table intolerable errors

|  |  |
| --- | --- |
| Error Type | Solution |
| Invalid map file | Report error, and ask for a correct map file. |
| Incomplete map without PLACES |
| Incomplete map without the “GDF…” line. i.e. no magic word, version number and environment name |
| Place or Artifact (including Key and Light) has no name |
| Duplicate ID within any section (excludes PATHS and LIGHTING where not tested) |
| Invalid source Place ID, destination Place ID, or direction in Path |
| Invalid location Place ID in Artifact |
|  |
|  |

Table 2 tolerable errors

|  |  |
| --- | --- |
| Error type | Solution |
| The number of staff stated in each section is inconsistent with real number.  e.g. nPlaces != places.size(), nArtifacts != artifacts.size(), … | Ignored. |
| Incomplete map with OPTIONAL sections missing |
| Invalid lightLevel value in LIGHTING, i.e. lightLevel beyond 0~100 | That line ignored. |
| Invalid Place ID in LIGHTING. |

# Lighting design

By default, the room light level is 0 before 6am or after 8pm, it’s 25 before 7am or after 7pm, otherwise it is 50.

Items are visible only when the light level in the room (including the usage of Lights) is between 15 and 100. // for executing LOOK, GET, GO (going to previous place is allowed), USE (using previously used artifacts or artifacts in his inventory is allowed)

# LOOK command design

The LOOK command can come with or without Direction or Object, meaning:

* LOOK: look around the surroundings of the room, describe current place and all outgoing paths.
* LOOK Direction: describe the path in the specific direction.
* LOOK HERE: look inside the room, list all items here without detailed description.
* LOOK Object: describe the specific item in the room or in the inventory.

# USE Artifacts design

Artifacts are treated as non-consuming, thus remaining in the INVE after executing the USE command (unless DROPped by the player).

After USE-ing a key to lock/unlock a path, even if the character leaves that place or later DROPs the key, those paths remain locked/unlocked.

After USE-ing a light to turn it on, the player carries its lighting with him. So if the character goes to another room with light active, the light will contribute to the lighting in this new room.

If the character USEd the light to turn it on, and DROPped it in the room, it will continue lighting the room.

//designed by Nianzu Ma

# Parsing the numbers in the map file

Negative integer numbers are allowed to be used as staff ID, as long as it’s unique.