

ppMazeWalkerのインストール

インストール

- インストール作業はありません
- 解凍してできたフォルダを好きなところに置いてください

アンインストール

- フォルダ削除でアンインストールが完了します
- 本ソフトはフォルダ内のファイルしか操作しません

テキストエディタのインストール

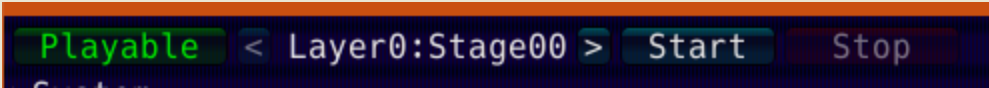
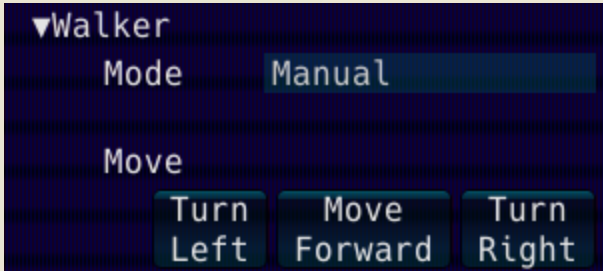
- VSCode等のプログラミング用テキストエディタをインストール
- プログラムが色分け表示されて見やすくなります

```
aaa = 10
bbb = "ccc"

if ddd == eee then
    fff.ggg(hhh, iii)
end
```

※ メモ帳でも遊べます

基本的な画面操作(マニュアル)

1. メニューバー	2. Mode を [Manual] にセット
	
[<] [>] ステージ選択	[TurnLeft] 左を向く
[Start] クエスト開始	[MoveForward] 1歩前進
クエストを開始したら2.へ	[TurnRight] 右を向く

基本的な画面操作(スクリプト)

1. メニューバー	2. Mode を [Script] にセット
	
[<] [>] ステージ選択	[Step >] 1回実行
[Start] クエスト開始	[Run ▷] 連続実行
クエストを開始したら2.へ	[Stop □] 連続実行停止

※ 実行前にスクリプトを編集&保存しておいてください

? Walker のプログラムはどこ？

(インストールフォルダ)/Walkers/MyWalker/Action.lua をエディタで開いてください。

```
(インストールフォルダ)
|
+--Walkers/
  +--_Don't_Use_/
  +--MyWalker
    +--Action.lua <== このファイル！！
    +--Start.lua
```

※ ファイルの拡張子が表示されるように設定しておいてください

? Walker を新しく作りたい

1. Walkers フォルダ内の MyWalker フォルダをコピー & リネームする
2. [Quest] を開き、Walker script を変更する

(インストールフォルダ)

```
|  
+--Walkers/  
    +--_Don't_Use_  
    +--MyWalker  
    |   +--Action.lua  
    |   +--Start.lua  
    +--MyWalker2      <== 追加した別のWalker(例)  
    |   +--Action.lua  
    |   +--Start.lua
```

? Walker プログラムで使えるコマンドを知りたい

[LocalVM仕様.pdf](#) を見てください。

Walkerプログラムがエラー ? で動かない !

1 Return文にヒットしないままプログラム最後まで流れてませんか？

どのif文も成立せず、return文に辿り着かないままプログラムの最後まで実行してしまうと、「return nil」を実行したときと同じ状態となります。

この場合、Walker の行動入力無しとなってクエストが中断されます。

どのような状況でも何かの行動を返すようにプログラムを調整してください。

Walkerプログラムがエラー？で動かない！

2 スペルミスしていませんか？

- 「if ~ then」が「if ~ yhen」のようにになっている
- 「return」が「retun」のようにになっている

Walkerプログラムがエラー ? で動かない !

3 文字列の「"(ダブルクォーテーション)」を付け忘れていませんか？

- 「return "TurnLeft"」とすべきところで「return "TurnLeft」 となっている、または「return TurnLeft」 となっている

Walkerプログラムがエラー？で動かない！

4 「.(ピリオド)」と「:(コロン)」を打ち間違えてませんか？

- 「his.prev_action」のようなプロパティ参照のときは「.(ピリオド)」で繋いでください
- 「map:Get(0, 0)」のような関数呼び出しのときは「:(コロン)」で繋いでください

※ 今はそういう決まりということで進めてください

? 画面左上にエラーが !

Error Reload Rule&Theme

ルール, テーマ, ステージスクリプトのどこかで問題が起きるとエラー表示になります。

- [Reload Rule&Theme] ボタンを押してください
- エラーが消えない場合は再インストールしてください
 - ルールかテーマのファイルが壊れている可能性があります

※ 再インストールの前に、作成したプログラムを避難させておきましょう

Layer1 の最後までクリアしてしまった.....

おめでとうございます !

完成したプログラムの単純さ(if文しか無いやんか！)に拍子抜けしたでしょう。ですが、**判定の仕方**、**処理の順序**の重要性は感じ取ってもらえたかと思います。

さて、基本問題として用意している Stage はこれで全部ですが、Layer1の Stage7 を使った応用問題を幾つか考えてみました。興味のある方はチャレンジしてみてください。

 応用問題も終わったらLayer2へ !!

▶ 1. 行き止まりのマスをスルーする

難易度(★☆☆☆☆)

アイテムが無く三方が壁になっている行き止まりをスルーするような機能を追加する。

必要となる知識

なし

▶ 2. 行動履歴機能(his)を自作する

難易度(★★☆☆☆)

プリセットされている his を使用せずに、同等の機能を自作して Walker を動かす。

必要となる知識

- Lua言語 ... 関数, 変数
- ppMazeWalker ... Start.lua の利用(初期化で必要となります)

▶ 3. 行動を予約できるようにする

難易度(★★★★☆)

次回, 次々回の行動を予め予約でき、古い予約から取り出して行動する機能を追加する。

必要となる知識

- Lua言語 ... 関数, 変数, for文, 配列
- ppMazeWalker ... Start.lua の利用
- プログラム ... キュー(Queue)

▶ 4. 高性能な行き止まりスルー機能

難易度(★★★★★)

- オートマッピング機能, 行き止まりを通行不要と認識する機能
- 通行不要に囲まれたところも通行不要と認識する機能

必要となる知識

- Lua言語 ... 関数, 変数, for文, 配列, テーブル
- ppMazeWalker ... Start.lua の利用

※これが出来たら一般的な迷路探索系プログラムとしてほぼ終点じゃないでしょうか