

シミュレーションライブラリについて

目次

1. はじめに
2. ディレクトリ構成
3. 概要
4. 要望・連絡

1. はじめに

このライブラリは、様々な食堂の混雑改善を目的とした Python のシミュレーションライブラリとなっています。それぞれの環境に合わせたシミュレーションが可能となっており、各自で追加することができます。また、完全なるオープンソースライブラリであり、誰でも気軽に貢献することもできます。まだまだ発展途上のライブラリですがよろしくお願いいたします。

2. ディレクトリ構成

このライブラリは、以下の構成でできています。

cafeteria-simulation (lib)

└ cafe	-> メインディレクトリ
├ └ __init__.py	-> 設定ファイル
├ └ core.py	-> メインファイル
├ └ example.py	-> Basic アルゴリズムと食堂のデータ
└ sample	-> sample ディレクトリ
├ └ _random.py	-> ランダムアルゴリズム
├ └ greedy.py	-> 貪欲法
├ └ puzzle.py	-> パズルアルゴリズム
├ └ basic.py	-> ベーシックアルゴリズム
└ tests	-> テスト用ディレクトリ

	└ test_simulation.py	-> テストファイル
	└ setup.py	-> 登録設定ファイル
	└ requirements.txt	-> 環境設定用ファイル
	└ README.md	-> ドキュメント
	└ LICENSE	-> ライセンス

3. 概要

3.1 変数・関数の説明

Cafeteria

Class

食堂のシミュレーションライブラリ

Cafeteria(data, time)で初期化を行う

・ data

食堂のデータ

・ data[0]

型: int

食堂のテーブルの数

・ data[1]

型: list of int

食堂のテーブル毎の席の数

・ data[2]

型: list of int

席の状態を座標で表した配列

・ time

型: int

シミュレーションを行う時間

・ table

型: int

食堂のテーブル数

・ number

型: int

テーブル毎の席の数

・ seats

型: list of int
食堂のテーブル数

- **group_member**

型: collection.deque of int
10 ターン後までに来るグループの人数

- **score**

型: list of int
各ターンのシミュレーションの合計スコア

- **flag**

型: int
現在のターンで座れた人数

- **index**

型: int
現在のターンで座れた人数

- **sum_penalty**

型: list of int
ペナルティ毎の累計

- **run(group)**

ターン毎のシミュレーションを実行する

- **group**

型: list of int
食堂のデータ

- **show()**

シミュレーション結果の図を表示する

- 得点の推移
 - ペナルティの個数
 - ペナルティの内訳

Toyota

Class

豊田高専の食堂のデータ
Cafeteria(data, time)で初期化を行う
Cafeteria(TOYOTA.data, time)のように扱う

basic_algorithm(next_member, seats, number)

利用者を模倣した着席アルゴリズム

・ next_member

型: list of int

着席させるグループの座標

・ seats

型: list of int

食堂のテーブル数

・ number

型: list of int

テーブル毎の席の数

3.2 利用方法

1. Pip コマンドを用いてインストールしてください
\$ pip install cafeteria-simulation
2. Python ファイルを作成
3. Import を行い, 初期値を設定します
4. ターン毎にグループの人数が送られてくるので席の座標を指定してください
5. 最後に show() を行うことでシミュレーション結果の図が表示されます

3.3 設定について

- ・ 1 ステップを 10 秒としています
- ・ 来る人数は 1~7 人です
- ・ 滞在時間は最短で 60 ステップ, 最大で 120 ステップとなっており, 食堂内での移動時間も滞在時間に含めています
- ・ 来る人数や滞在時間はランダムです.
- ・ 比較ができるようにランダムの SEED 値を設定しています
- ・ 座ることで 100 点加算, ペナルティを受けることで減点します
- ・ Basic アルゴリズムでの点数を基準値として改善率を求めています

4. 要望・連絡

このライブラリは色々な方の支援によって作られています！要望などありましたら，このライブラリを管理している **Github** のサイトでプルリクエストを送信したり，下記のメールに送って頂きたいです！

メールアドレス： 31807@toyota.kosen-ac.jp