

プログラミング 8th 「モデル化とシミュレーション」  
～モデル化とシミュレーションによる問題解決～

## 1. 目標

『フォーク並び』と『各レジ並び』、どちらが平均待ち時間や公平性において優れているか、シミュレーションで検証する。

## 2. 本時の流れ

①はじめに（本時の目標）→②モデルの理解と変数設定→③グループワーク：シミュレーション実行  
→④共有・考察→⑤まとめ

## 3. モデルの理解と変数設定

### ○入力（変数）

- ①レジの台数 `NUM_REGISTERS = 3` # レジの台数
- ②レジの処理速度(平均接客時間) `AVG_SERVICE_TIME = 3` # 平均接客時間（分）
- ③客が来る間隔 `ARRIVAL_INTERVAL = 1` # 客が到着する間隔の平均  
(ポアソン分布のλ値として使用)

### ○出力（評価指標） ①平均待ち時間、②最大待ち時間

### ○不確定要素（乱数）

「前の客が小銭を出すのに手間取る」、「クーポンを探す」などのばらつき。

```
service_times = np.random.poisson(AVG_SERVICE_TIME, NUM_CUSTOMERS) + 1  
# 最低1分以上
```

◎サービス時間、理論上の最大値 ※ここはグループ活動のときにポイントになるよ  
ポアソン分布は離散確率分布の一つで、理論上は無限大まで値を取る可能性があります。  
しかし、実際に大きな値が出る確率は急激に低くなります。

今回の設定（平均  $\lambda = 3$ ）の場合、各値が出る確率は以下のようになります：

- ・1分 (ポアソン 0 + 1): 約 5.0%
- ・4分 (ポアソン 3 + 1): 約 22.4% (ここがピーク)
- ・10分 (ポアソン 9 + 1): 約 0.27%
- ・15分 (ポアソン 14 + 1): 約 0.0001%

つまり、理論上は上限がありませんが、現実的なシミュレーション（客数 50 人程度）では、10分～12分程度が実質的な最大値になることがほとんどです。

#### 4. グループワーク：シミュレーション実行

<シミュレーター>

Google Colaboratory 【プロ 8th】 待ち行列\_モデル化とシミュレーション.ipynb

<ワーク①>

- ・「客が1分に1人来る」という条件で、フォーク並びと各レジ並びを20回分試行する。
- ・それぞれの「平均待ち時間」「最大待ち時間」を記録し、特徴と傾向を捉える。

→ スプレッドシート 【入力】 各班の試行結果入力用\_プロ 8th に入力する

<ワーク②>

- ・「もし、1つのレジだけベテラン店員で爆速だったら？」
- ・「もし、大量に買う客が混じっていたら？」と条件を変えてみる。

→ スプレッドシート 【探究】 条件変更の比較\_プロ 8th に入力する

#### 5. ルーブリックの評価項目とそのレベル

※授業の最後に個人評価をしてもらいます。事前に「評価項目とそのレベル」を確認してください。  
本時の学習は、このことを意識して取り組みましょう。

評価項目	レベル1 (C)	レベル2 (B)	レベル3 (A)	レベル4 (S)
思考・判断	入力(客の数など)と出力(待ち時間)の関係が理解できず、シミュレーションの結果を単なる数値として受け取るだけにとどまる。	入力と出力の関係を理解し、提示された条件でシミュレーションを行い、平均待ち時間の違いなど、シミュレーション結果の事実を正しく読み取れる。	変数(処理速度や客数)を自分で調整し、結果がどう変わるか論理的に推測でき、数値的な優劣(平均・最大)だけでなく、不公平感や格差といった視点で結果を分析できる。	現実の複雑な要素(レジの習熟度、客の荷物量など)を独自の変数として定義し、モデルに反映でき、数値データに加え、店舗面積や顧客心理などの非数値的要素を組み合わせ、多角的な解決策を提案できる。