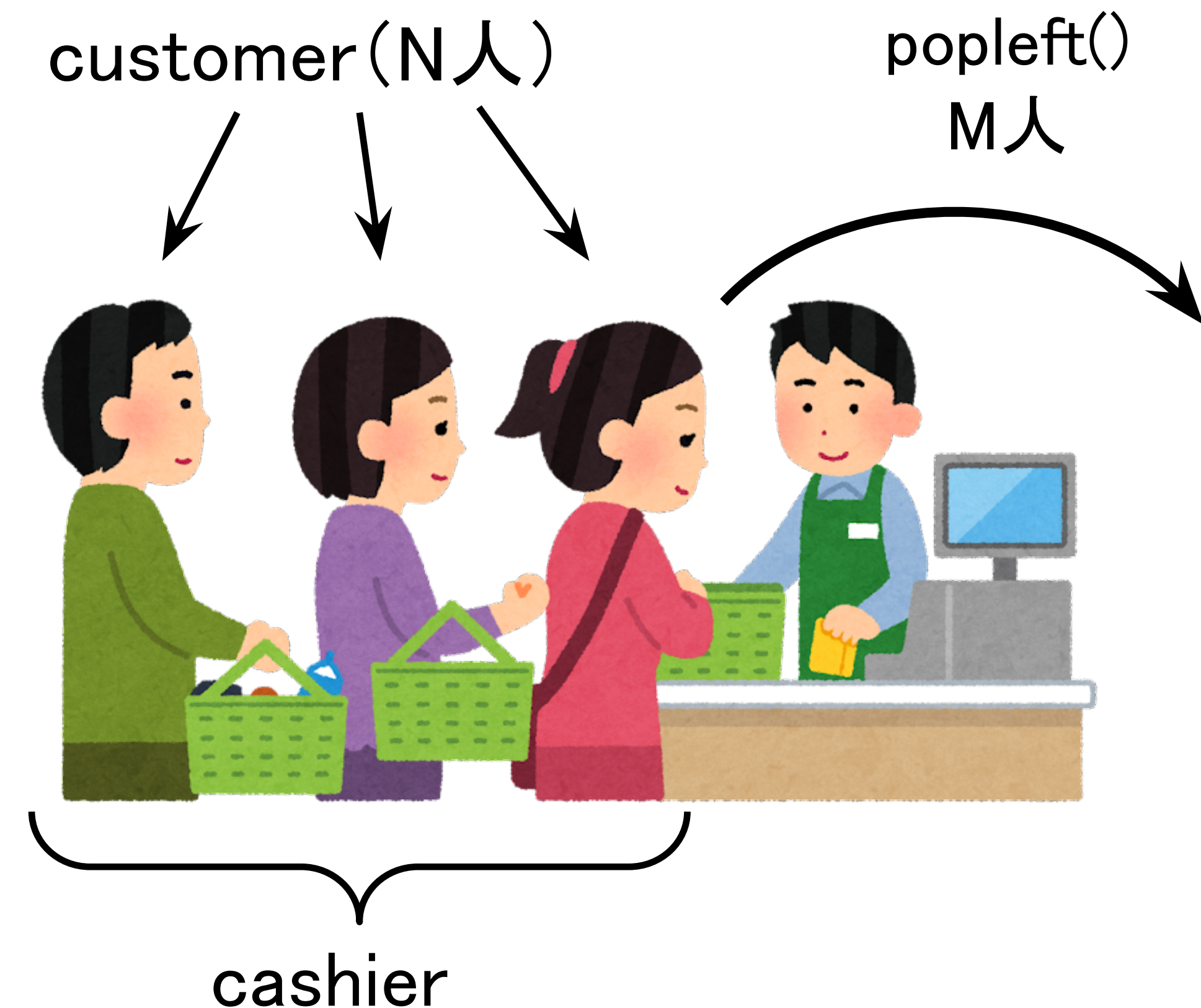


## 課題1:

## レジの待ち行列シミュレーション(N人バージョン)

- 先ほどは、2人のお客さん(A, B)がレジに並び、2人が会計を済ませる状況をシミュレーションしました。
- ここでは、N人のお客さんがレジに並び、その後M人がレジで会計を済ませる状況を考えましょう。
- 次のコードの空欄(A)~(I)を埋めて、ソースコードを完成させましょう。
- (A)~(I)に当てはまるものを提出してください。
- ※ $M > N$  の場合もうまく動作するようにしましょう。



# ソースコードの例: (A)~(I)を埋めよう

## Markdown

```
### 課題1 レジの待ち行列シミュレーション(N人バージョン)
```

## Code

```
from ____ (A) ____ import ____ (B) ____

# レジの待ち行列を作成
cashier = ____ (C) ____

n = 10    # レジに並ぶお客さんの数
m = 6     # お会計を完了するお客さんの数

for i in ____ (D) ____: # n人の客が並ぶループ
    customer = "customer-id:{}".format(i)    # お客さんのID
    print(customer, "さんがレジに並びました。")
    cashier.____ (E) ____

for i in ____ (F) ____: # m人の客が会計を済ませるループ
    if ____ (G) ____:
        customer = cashier.____ (H) ____
        print(customer, "さんが会計を済ませました。")

# レジに並んでいる人の数を入力する
print("レジに並んでいる人数:", ____ (I) ____)
```

## 実行結果

```
customer-id:0 さんがレジに並びました。
customer-id:1 さんがレジに並びました。
customer-id:2 さんがレジに並びました。
customer-id:3 さんがレジに並びました。
customer-id:4 さんがレジに並びました。
customer-id:5 さんがレジに並びました。
customer-id:6 さんがレジに並びました。
customer-id:7 さんがレジに並びました。
customer-id:8 さんがレジに並びました。
customer-id:9 さんがレジに並びました。
customer-id:0 さんが会計を済ませました。
customer-id:1 さんが会計を済ませました。
customer-id:2 さんが会計を済ませました。
customer-id:3 さんが会計を済ませました。
customer-id:4 さんが会計を済ませました。
customer-id:5 さんが会計を済ませました。
レジに並んでいる人数: 4
```