

Wstęp

W tym etapie rozszerzamy nasz program o obsługę różnych rozmiarów skrytek i paczek

Wymagania

- Nasz paczkomat obsługuje teraz paczki o rozmiarach S;M;L
- Do skrytki danego rozmiaru pasuje paczka o rozmiarze conajwyżej rozmiaru skrytki, dlatego do skrytki:

```
S -> wejdzie tylko paczka o rozmiarze S
M -> wejdzie paczka o rozmiarze S lub paczka o rozmiarze M
L -> wejdzie paczka o rozmiarze S lub paczka o rozmiarze M lub paczka o rozmiarze L
```

- Daną paczkę wkładamy do pierwszej skrytki o minimalnym rozmiarze jaką znajdziemy
- Rozszerzamy tworzenie paczkomatu o rozmiary skrytek i ich zajętość, przykładowo

```
|S:S S:O S:O M:X M:M|
|L:O L:O S:X M:S L:S|
```

gdzie

S:X lub *S:S* oznacza skrytkę S zawierającą paczkę S

S:O oznacza małą skrytkę pustą

M:X lub *M:M* oznacza skrytkę M zawierającą paczkę M

M:O oznacza średnią skrytkę pustą

L:X lub *L:L* oznacza skrytkę L zawierającą paczkę L

L:O oznacza dużą skrytkę pustą+

- Po stworzeniu paczkomatu można podać dowolną liczbę razy komendę dodającą paczki do paczkomatu w formie:

```
BM;BS;BL;B
```

gdzie

BS lub *B* oznacza akcję dodania nowej paczki małej

BM oznacza akcję dodania nowej paczki średniej

BL oznacza akcję dodania nowej paczki dużej

- Wkładanie paczek obsługujemy w kolejności od lewej do prawej
- Przy wypisaniu paczkomatu - paczki pasujące rozmiarem do skrytki wypisuje jako "X" a nie po identyfikatorze rozmiaru, przykładowo

```
|S:X M:X L:X|
```

zamiast

|S:S M:M L:L|