

## Zad 8.

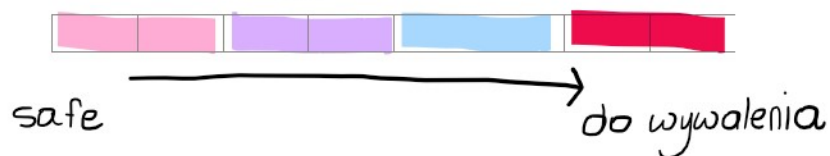
poniedziałek, 29 maja 2023 14:34

**Zadanie 8.** Dla czterodrożnej sekcyjno-skojarzeniowej pamięci podręcznej implementujemy politykę zastępowania LRU. W każdym zbiorze przechowujemy maksymalnie 8 dodatkowych bitów (nazwijmy je *state*), które kodują listę wierszy od ostatnio używanego do najdawniej używanego. Funkcja *victim* : *state* → *line* służy do wyznaczania wiersza ofiary, a funkcja *update* : *state* × *line* → *state* do aktualizacji stanu wiersza, jeśli procesor wygenerował trafienie w wiersz *line*. Podaj implementację funkcji *victim* i *update* w języku C używając wyłącznie operatorów bitowych, ale nie arytmetycznych! Wyłumacz jak kodujesz listę wierszy przy pomocy *state*.

**Wskazówka:** W stanie wiersza zakoduj kolejność elementów w wierszu.

**Zastanów się:** Czemu zależy nam na zminimalizowaniu liczby bitów kodujących *state*, a nie zminimalizowaniu logiki realizującej funkcję «victim» i «update»? *zastanowiłem się i nie wiem*

*kodowanie → dzielimy 8 bitów na 4 pary*



*w parach kodujemy indeksy wierszy, czyli np*

*00 | 01 | 11 | 10*

*↖ wiersz 2 jest kolejny do wywalenia*

```
1. uint8_t victim(uint8_t state) {
    return (state & 0b11);
}
```

```
2. uint8_t update(uint8_t state, uint8_t line) {
```

