

ALGORYTMY I STRUKTURY DANYCH

KOSZT ZAMORTYZOWANY I TABLICE DYNAMICZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

1. **[**]** Jaki jest całkowity koszt wykonania na stosie ciągu n instrukcji *push*, *pop* i *multipop*? Zakładamy, że na stosie znajdowało się początkowo $s_0 > 0$ a na końcu s_n elementów. Zastosuj *metodę kosztu sumarycznego* do wyznaczenia zamortyzowanego czasu działania pojedynczej instrukcji w tym ciągu.
2. **[**]** Pokaż, jak zaimplementować kolejkę za pomocą dwóch zwykłych stosów, aby zamortyzowany koszt operacji *enqueue* i *dequeue* był rzędu $O(1)$. Zastosuj *metodę księgowania*.
3. **[**]** *Licznik binarny* to struktura danych reprezentowana przez k -elementową tablicę bitów $B[0 \dots k-1]$. W tablicy tej pamiętana jest nieujemna wartość całkowita w postaci binarnej, zwana licznikiem. Na liczbie tej można wykonywać jedynie operację **increment()** zwiększającą wartość licznika o 1. Operacja ta zaprogramowana jest w oparciu o pisemne dodawanie jedynki do liczby binarnej:

```
procedure INCREMENT (bit  $B[0 \dots k-1]$ )  
{  
    for  $i = 0 \dots k-1$  do  
        if  $B_i = 1$   
            then  $B_i \leftarrow 0$ ;  
            else {  $B_i \leftarrow 1$ ; return; }  
}
```

Określ faktyczny czas działania pojedynczej operacji **increment()**. Przeanalizuj czas wykonania ciągu n takich operacji na początkowo wyzerowanym liczniku. Dokonaj analizy zamortyzowanej *metodą potencjału*.

4. **[**]** Przeanalizuj czas wykonania ciągu n operacji *append* i *cut* na początkowo pustej tablicy dynamicznej ze współczynnikiem powiększania $\kappa = 2$. Dokonaj analizy zamortyzowanej *metodą potencjału*. Jako funkcję potencjału przyjmij:

$$\Phi(T) = \begin{cases} 2 \cdot T.size - T.capacity & : \alpha(T) \geq 1/2 \\ T.capacity/2 - T.size & : \alpha(T) < 1/2 \end{cases}$$

5. **[***]** Jakich zmian należy dokonać, aby każda operacja (*append*, *cut* i *at*) na *tablicy dynamicznej* była wykonywana zawsze w czasie $O(1)$?