

Datastrukturer: linked lists, stakke, køer (v2)

Imperativ Programmering og Datastrukturer

René Rydhof Hansen

November 2007

Mål

- At kunne definere datastrukturen: linked lists (hægtet liste)
- At kunne forklare fordele og ulemper ved linked lists
- At kunne implementere og anvende linked lists

Arrays

Fordele

Ulemper

Fordele

- Hurtigt opslag på nøgleværdi
- Let at forstå
- Understøttes direkte af sproget

Ulemper

Fordele

- Hurtigt opslag på nøgleværdi
- Let at forstå
- Understøttes direkte af sproget

Ulemper

- Statisk størrelse (ikke helt rigtigt)
- Svært/dyrt at indsætte og slette elementer
- Svært/dyrt at sortere/vedligeholde sortering

Linked Lists

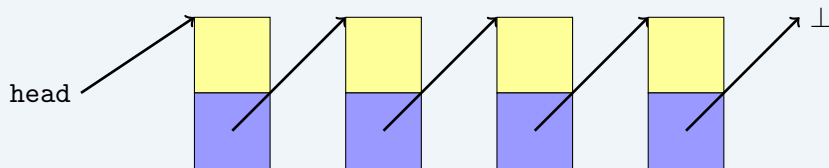
Example

Structure Record

```
Dim titel as String
```

```
Dim next as Record
```

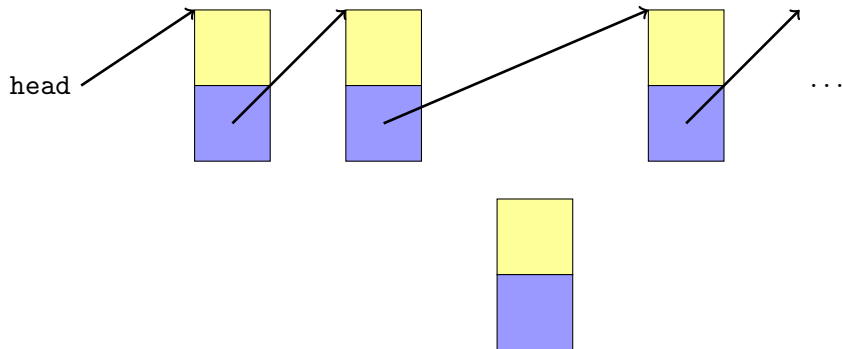
End Structure



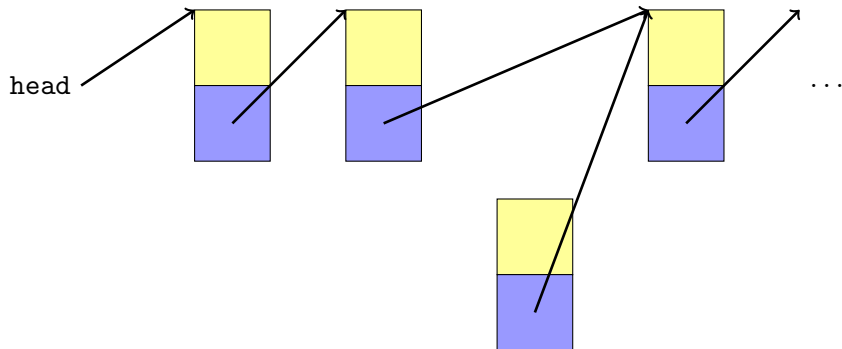
Operationer på Linked Lists

- Indsættelse af element
 - Head
 - Tail
 - Ordered
- Fjernelse af element
- (Reversering af liste)

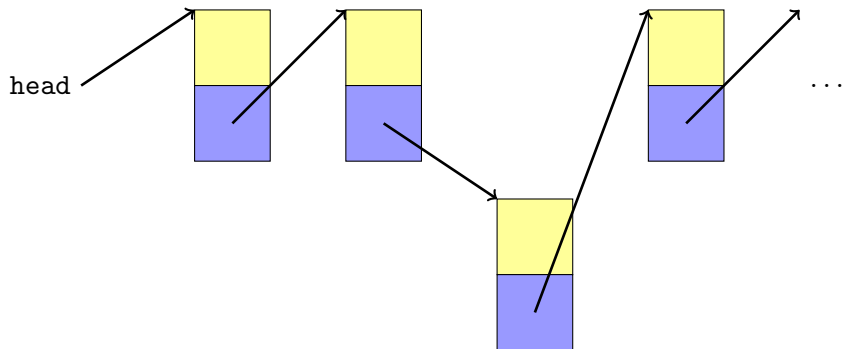
Indsættelse af element i en linked list



Indsættelse af element i en linked list



Indsættelse af element i en linked list



Linked Lists: Fordele og Ulemper

Fordele

- Dynamisk størrelse
- Let at indsætte/fjerne elementer
- Let at vedligeholde sortering
- Meget fleksibel

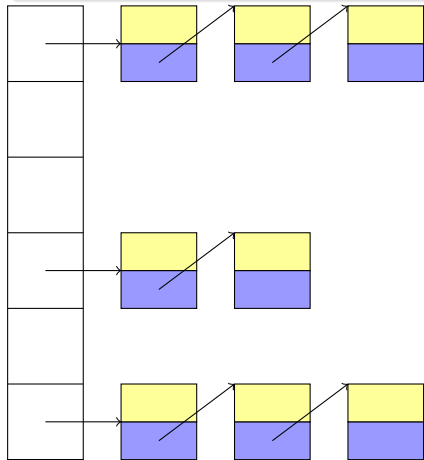
Ulemper

- Dyrere søgning
- Mere overhead (ingen indbygget sprog-støtte)
- Vanskelig at sortere *a posteriori*

Array af Linked Lists (Hash-tabeller)

Nej, *ikke* den slags hash

Fra engelsk: "to chop and mix" eller "a confused mess".



Stak

- Last In, First Out (LIFO) / First In, Last Out (FILO)
- Operationer
 - Push (indsæt top-element)
 - Pop (fjern top-element)

Kø

- Last In, Last Out (LILO) / First In, First Out (FIFO)
- Operationer
 - Put (indsæt nyt bund-element)
 - Get (fjern top-element)