

TDIU20

Tentaworkshop 2

Eric Ekström och Klas Arvidsson

Institutionen för datavetenskap

Upplägg

Första halvan:

- Gemensam genomgång
- Självständigt arbete
- Sammafattning

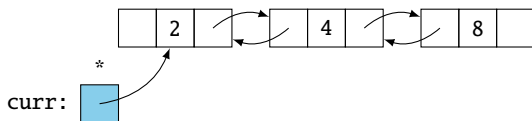
Andra halvan:

- Gemensam genomgång
- Självständigt arbete
- Sammafattning

- 1 Del 1 - Pekarmanipulation
- 2 Del 2 - Speciella medlemsfunktioner
- 3 Hemuppgifter

Genomgång - Länkad struktur

(Tentauppgift från 2024-08)



Du har ovanstående struktur för en lista. Skriv kod som sätter in ett nytt värde 6 mellan de existerande elementen med värde 4 och 8. Det du har tillgång till är variabeln `curr`. Antag noder av typ `struct Node{ Node* prev; int val; Node* next; }.`

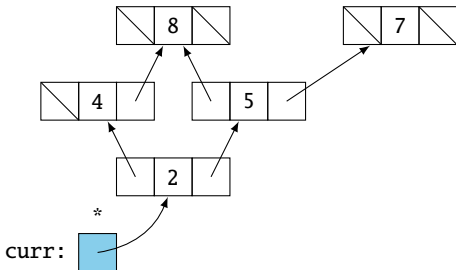
KRAV: Du får inte byta plats på värden, endast flytta pekare. Samtliga element är dynamiskt allokerade och koden får inte generera minnesläckor.

Genomgång - Skapa en länkad struktur

(Tentauppgift från 2025-03)

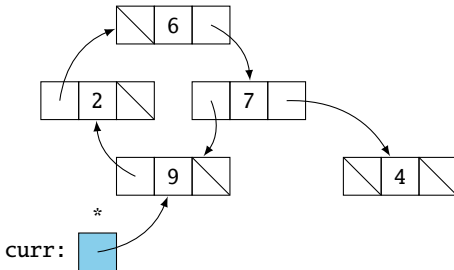
Skapa följande länkade stuktur. Alla element ska vara dynamiskt allokerade. Antag noder av typ:

```
struct Node
{
    Node* prev;
    int const val;
    Node* next;
};
```



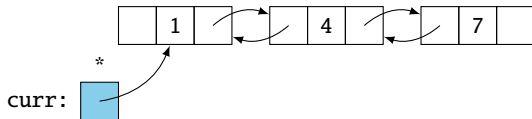
Lös själv - Skapa en struktur

Skriv kod som skapar den länkade strukturen nedan.
Alla element ska vara dynamiskt allokerade.



```
struct Node
{
    Node* prev;
    int const val;
    Node* next;
};
```

Lös själv - Modifiera en struktur



Utgå från strukturen ovan och

- ta bort noden med värde 4
- sätt in en node sist i strukturen
- byt plats på noderna med värde 4 och 7

Försök använd så få extra variabler som möjligt. Skapa strukturen först och skriv sedan separata program för varje punkt.

- 1 Del 1 - Pekarmanipulation
- 2 **Del 2 - Speciella medlemsfunktioner**
- 3 Hemuppgifter

Genomgång - Speciella medlemsfunktioner

- Kopieringskonstruktor
X(X const&)

Genomgång - Speciella medlemsfunktioner

- Kopieringstilldelningsoperator
`X& operator=(X const&)`

Genomgång - Speciella medlemsfunktioner

- Flyttkonstruktör
 $X(X \ \&\&)$

Genomgång - Speciella medlemsfunktioner

- Flyttilldelningsoperator
`X& operator=(X &&)`

Genomgång - Speciella medlemsfunktioner

- Destruktor
 $\sim X()$

Genomgång - Speciella medlemsfunktioner

- Kopieringskonstruktor
`X(X const&)`
- Kopieringstilldelningsoperator
`X& operator=(X const&)`
- Flyttkonstruktor
`X(X &&)`
- Flyttilldelningsoperator
`X& operator=(X &&)`
- Destruktor
`~X()`

Lös själv - Speciella medlemsfunktioner

Skriv en klass som implementerar alla fem speciella medlemsfunktionerna. Beteendet för varje speciell medlemsfunktion är att de skriver ut ett meddelande i terminalen (tex "Kopieringskonstruktor!").

Skriv ett testprogram som tydligt visar på att varje unik utskrift sker i terminalen.

(Extra utmaning: lös detta utan att använda `std::move`).

(Går det att få ut varje utskrift exakt 2 gånger?)

- 1 Del 1 - Pekarmanipulation
- 2 Del 2 - Speciella medlemsfunktioner
- 3 Hemuppgifter

Gör hemma - Pekarmanipulation

Skriv kod för att destruera de strukturer vi byggt tidigare. Din lösning behöver inte vara generell, utan ska fungera specifikt för just den strukturen.

Som vidare övning, testa att skapa en länkad struktur likt de ovan (dubbellänkade noder som sitter ihop i ett intressant mönster), modifiera den och frigör till sist minnet.

www.ida.liu.se/~TDIU20/