

# Datastrukturer och algoritmer

## Föreläsning 4

77

## Projekt

- Definition: En grupp av projektdeltagare utför under ledning av en projektledare en klart definierad uppgift, på en viss tid, med begränsade resurser
- Resurserna kan vara i form av människor, material, pengar, eller lokaler
- Projekt ska ha mätbara mål

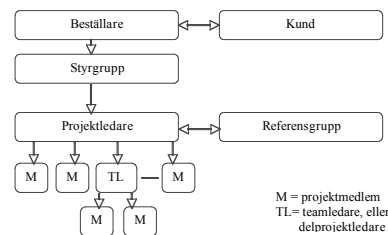
78

## Projekt

- Ett projekt är en engångsföreteelse!
  - Dokumentation är mycket viktigt
- Fördelar med projekt
  - Arbetsuppgifter kan utföras parallellt och därmed slutföras på kortare tid
  - Skapar arrangemang genom att arbetet utförs i en eller flera grupper
  - Projekt bemannas ofta med personer med kompletterande kunskaper

79

## Roller i ett industriellt projekt



M = projektmedlem  
TL = teamledare, eller delprojektledare

Bilden hämtad från projektplanemallen

80

## Roller

- Beställare
  - Kan vara en extern kund till ett företag eller en intern beställare (ex produktchef)
  - Ansvarig för nyttan med projektet, om det är värt investeringen
- Projektledare
  - Ansvarar för att projektets utförs och slutförs enligt givna direktiv
- Projektmedlem
  - kan vara ansvarig för att en aktivitet utförs
  - kan också utses till dokumentansvarig, kvalitetsansvarig, testansvarig, kundansvarig, designansvarig,...

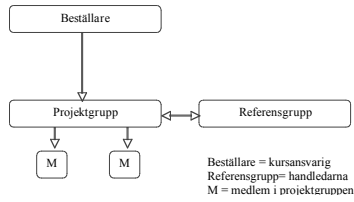
81

## Roller

- Styrgrupp
  - Utses av beställaren för att styra och bevaka att projektet når målen.
- Referensgrupp
  - Personer som stöttar projektledaren och projektmedlemmarna (utan beslutanderätt).
- Leverantör
  - Person/företag utanför projektgruppen som kontrakteras för att utföra arbete/leverera utrustning.

82

## Organisation i vårt projekt



83

## Projektplan

- Specifikation av hur projektet ska genomföras
- Dokumenterar till exempel
  - Mål
  - Resurser
  - Tidplan

84

## LIPS begrepp

- Milstolpar
  - En signifikant mätbar händelse
  - Etappmål
  - Definieras oftast av projektgruppen själv
- Beslutspunkter
  - Punkter där beställaren bestämmer om projektet får fortsätta in i nästa fas.
  - Ofta resultat vid en milstolpe som ligger till grund för beslut.
- Aktiviteter
  - De arbetsuppgifter som ska utföras under projektet och plan för tidsåtgången för var och en av dessa.
- Granskningar
  - Varje dokument måste granskas innan de godkänns.

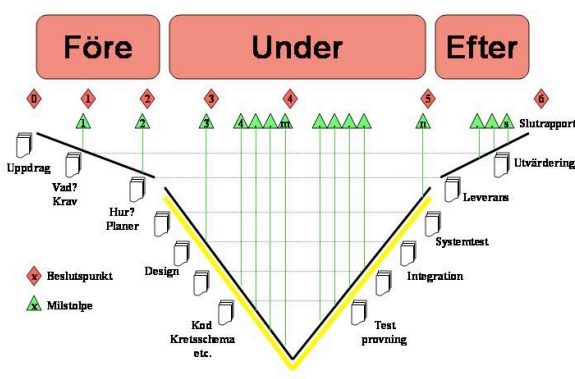
85

## Projektstyrningsmodell

- Regler och hjälpmedel för att bedriva ett projektarbete
- Gemensamma definitioner och beskrivningar av flöden, aktiviteter, roller, dokument, etc
- Varje företag har oftast en egen projektmodell
  - Konfidentiell (konkurrensmedel!)
  - Ericsson använder en modell som heter PROPS
  - Saab använder en modell som heter PSM
- Vi kommer att använda LIPS
  - Lätt Interaktiv ProjektStyrning

86

### LIPS

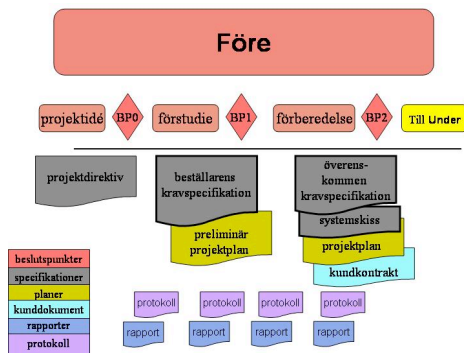


## Huvudfaser i LIPS - Före

- I denna fas ges uppdraget och utförandet planeras.
- En projektplan skapas
  - Kravspecifikation definierar vad man ska göra
  - Systemskiss visar hur man ska göra det
  - Aktiviteter identifieras
  - Resurser och tid planeras
- Viktig fas! Görs planeringen fel kan projektet inte bli lyckat.
- Tar tid men man går inte på djupet i detaljerna

88

bilden hämtad från: <http://www.liu.se/cul-resurser/lips/kartor/fore.htm>



89

## Före-fasen under kursen:

- Projektidé och förstudie har redan gjorts och BP0, MS1 och BP1 har passerats.
  - Dokumentet "Projektdirektiv" och "Kravspecifikation" finns på projektsidan.
- Ni får uppdraget (OU3) att förbereda projektet inför utförandefasen.
  - Kraven studeras och man beskriver hur man ska göra i en systemskiss/funktionsspecifikation.
  - Projektplan upprättas
- MS2 består av projektplan och kravspecifikation
- BP2 är rättningen av OU3
  - endast U medför att projektet ej får fortsätta

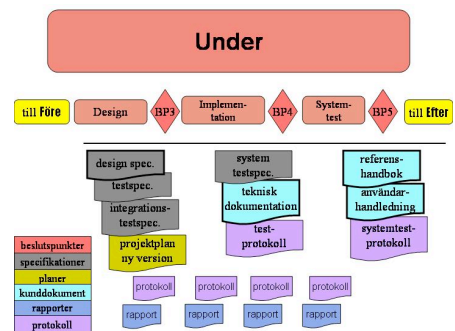
90

## Huvudfaser i LIPS - Under

- Utför projektet och dess aktiviteter
  - Detaljerade specifikationer skapas
  - Aktiviteterna utförs, dokumenteras och resultatet testas.
  - Kontinuerlig rapport till beställaren (mötesprotokoll och statusrapporter)
  - Fasen avslutas med ett systemtest.
- Arbetet går i cykler
  - Upptäcka att ett krav påverkar designen, måste göra ny design tex.

91

Bilden hämtad från: <http://www.liu.se/cul-resurser/lips/kartor/under.htm>



92

## Under-fasen under kursen:

- Testplan där man funderar på vad som ska testas och hur är viktigt för att garantera fungerande slutresultat.
- Viktigt att lägga in många milstolpar och stämma av tidsplan och testplan för att se eventuella problem tidigt.
- Man kan behöva revidera projektplanen.
- Krav under kursen
  - Utföra individuell tidsrapportering
  - Redovisa pågående arbete och reviderad projektplan (OU5).
    - Milstolparna ni beslutar er för att ha ska synas i projektplanen
- Vi använder endast BP5 som är rättningen av OU5.

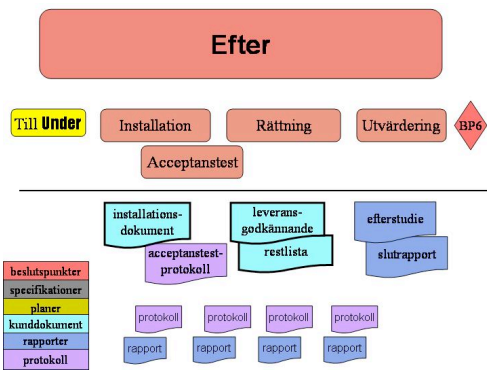
93

## Huvudfaser i LIPS - Efter

- Projektresultatet överförs till beställaren och projektet avslutas.
- Utvärdering utförs.
- Ofta det svåraste i ett projekt... Att få det betalt och avslutat!

94

Bilden hämtad från: <http://www.liu.se/cul-resurser/lips/kartor/efter.htm>



## Efter-fasen under kursen:

- Här lämnar ni in slutversionen tillsammans med dokumentation av systemet och efterstudien direkt (OU6).
- Handedarna utför acceptanstestet utifrån kravspecifikationen och kan komma med en restlista...
- BP6 är med andra ord rättningen av OU6

96

## Hashtabell

- Problemen med att implementera en mängd/lexikon som en bitvektor är att grundmängden inte får bli för stor och att många små mängder slösar med minnet.
  - Vore bra att kunna komprimera bitvektorerna.
- Vi behöver en *hashfunktion* som avbildar den större indextypen A till en mindre indextyp B.
- Man kan också se det som att grundmängden är nycklarna i en tabell.
  - Eller att man vill omvandla nycklar i en tabell till siffror med hjälp av hashfunktionen.

97

## Hashfunktion

- Funktionen  $f(x)$  avbildar en stor mängd nycklar, A, på en liten mängd tal, B.
- Kollisioner är oundvikliga och en bra hashfunktion kännetecknas av:
  - Litet förväntat antal kollisioner
  - Sprider elementen jämt över mängden.
  - Bör påverkas av alla delar av nyckeln
- Kollisioner kan hanteras med
  - Sluten hashing
  - Öppen hashing

98

## Sluten hashing

- Låt B vara en cirkulär vektor, dvs efter det sista elementet följer det första.
- Linjär teknik för kollisioner
  - Om ett element kolliderar (platsen redan upptagen) sätter man in det på den första lediga platsen efter den upptagna.
    - Ger upphov till klustring i tabellen.
  - Sökning efter ett element börjar på den plats dess hashvärde anger och fortsätter eventuellt framåt. Om det inte påträffats före nästa lediga plats så finns det inte i tabellen.
  - Om vi vid borttagning i tabellen bara lämnar platsen tom blir det fel vid sökning.
    - Sätt in en "borttagen" markör i tabellen.

99

## Exempel: Sluten hashing, linjär teknik

- Grundmängden  $1, 2, \dots, 125$  ska avbildas på värdena  $0, 1, \dots, 6$ .
- Mängd =  $\{1, 8, 27, 64, 125\}$   
Använd hashfunktionen  $f(x) = x \% 7$ .

0		0		0		0		0	125				
1	1	1	1	1	1	1	1	1	1				
2	$h(1)$	2	8	$h(8)$	2	8	$h(27)$	2	8	$h(64)$	2	8	$h(125)$
3	= 1	3		= 1	3		= 6	3	64	= 1	3	64	= 6
4		4		4		4		4		4		4	
5		5		5		5		5		5		5	
6		6		6		6	27	6	27	6	27	6	27

100

## Sluten hashing, linjär teknik

- Värstafallskomplexiteten för samtliga operationer är  $O(n)$  där  $n$  är antalet element som finns insatta i tabellen.
  - Är dock ytterst osannolikt. Alla element måste ligga i en följd.
- Under förutsättning att tabellen inte fylls mer än till en viss del får man i medeltal  $O(1)$  för operationerna.

101

## Hashtabeller - fyllnadsgrad

- En hashtabells *fyllnadsgrad* ( $\lambda$ ) definieras som kvoten mellan antalet insatta element och antalet platser i tabellen.
  - En tom tabell har  $\lambda = 0$  och en full  $\lambda = 1$ .
- Man vet att
  - Medelantalet platser som måste prövas vid en insättning och misslyckad sökning är uppåt begränsad av  $(1+1/(1-\lambda)^2)/2$ , ( $\lambda = 0.5$  ger 2.5)
  - Medelantalet platser för en lyckad sökning är uppåt begränsad av  $(1+1/(1-\lambda))/2$ , ( $\lambda = 0.5$  ger 1.5)

102

## Sluten hashing, kvadratisk teknik

- När fyllnadsgraden blir hög (>75%) blir operationerna för den linjära tekniken långsam.
- Kvadratisk teknik kan användas istället:
  - Vid kollision tas först nästa element, sedan det fyra platser fram sedan nio fram etc, dvs  $H, H+1, H+4, \dots, H+i^2, \dots$  där  $H$  är elementets hashvärde.

103

## Exempel: Sluten hashing, kvadratisk teknik

- Sätt in talen 89, 18, 49, 58, 9 i en tabell med 10 platser.  $h(x) = x \% 10$

	0	1	2	3	4	5	6	7	8	9
89 % 10 = 9										
18 % 10 = 8										
49 % 10 = 9, upptagen										
58 % 10 = 8, upptagen										
9 % 10 = 9, nästa lediga plats										

104

## Sluten hashing – Kvadratisk teknik

- Inte säkert att man hittar en ledig plats även om det finns! Exempel:
  - Man har en tabell som är 16 element stor och man använder hashfunktionen  $h(x) = x \% 16$
  - Stoppar in elementen 0, 16, 32 och 64. Då finns det inte plats för några fler tal som hashas till 0. De enda platser som kommer prövas är de upptagna 0, 1, 4, 9.
- Men:
  - Om kvadratisk teknik används och tabellens storlek är ett primtal så kan ett nytt element alltid stoppas in om fyllnadsgraden  $< 0.5$ .
  - Fullständig komplexitetsanalys saknas men i praktiken ger den upphov till mindre klustring än linjär hashing.

105

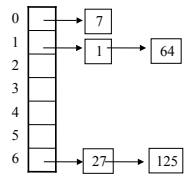
## Öppen hashing

- Istället för cirkulär vektor används en vektor av lista. I lista nummer  $i$  ligger alla element med hashvärde  $i$ .
- Värstafallskomplexitet:
  - $O(n)$  för alla operationer (Alla element i samma lista).
- Medelfallskomplexitet:
  - Insättning och misslyckad sökning blir  $n/\text{tableSize}$
  - Lyckad sökning blir  $(n-1)/(2*\text{tableSize})+1$
- Tumregel: Inte fler än  $2*\text{tableSize}$  element bör sättas in.

106

## Exempel: Öppen hashing

- Grundmängden 1, 2, ..., 125 ska avbildas på värdena 0, 1, ..., 6.
- Mängd = {1, 7, 27, 64, 125}
- Använd hashfunktionen  $f(x) = x \% 7$ .



107

Från: <http://www.ida.liu.se/labs/logpro/ulfni/dalg/resources/f8.pdf>

## Teoretiska resultat

Antal sonderingar vid olika fyllnadsgrad:

**Linear prob** = Sluten hashing med linjär teknik  
**Chaining** = Öppen hashing  
**Double hash** = Hashning med en teknik som liknar den slutna med kvadratisk teknik

	Fyllnadsgrad (%)					
	10	25	50	75	90	99
	Lyckad sökning					
chaining	1.05	1.12	1.25	1.37	1.45	1.49
linear prob.	1.06	1.17	1.50	2.50	5.50	50.5
double hash	1.05	1.15	1.39	1.85	2.56	4.65
	Misslyckad sökning					
chaining	0.1	0.25	0.5	0.75	0.9	0.99
linear prob.	1.12	1.39	2.50	8.5	50.5	5000
double hash	1.11	1.33	2.0	4.0	10.0	100

108

## Hashfunktionen

- Vi har använt  $h(x) = x \% m$  i exemplen.
  - Fungerar bra om  $m$  är ett primtal.
  - Annars kan periodicitet uppstå.
- Om  $x$  tal så finns mer generella metoder, tex
  - $h(x) = ((c_1 * x + c_2) \% p) \% m$ 
    - $p$  stort primtal  $> m$ , tex 1048583
    - $c_1$  och  $c_2$  konstant heltal  $> 0$  och  $< p$
    - $c_2$  ofta satt till 0

109